5-30-2017

Performance Evaluation of MPLS in a Virtualized Service Provider Core (with/without Class of Service)

Utkarsh Shah
uhs6220@rit.edu

Master of Science in

Applied Networking and System Administration

Thesis Approval Form

Student Name: Utkarsh Shah

Thesis Title: Performance Evaluation of MPLS in a Virtualized Service Provider Core (with/without Class of Service)

Thesis Area: ✓Networking ☐ Systems Administration ☐ Security ☐ Other _____

MS Thesis Committee

Name	Signature	Date

Dr. Ali Raza
Chair

Dr. Charles Border
Committee Member

Dr. Harry Manifavas
Committee Member

R.I.T

Performance Evaluation of MPLS in a Virtualized Service Provider Core (with/without Class of Service)

by

Utkarsh Shah

Committee Members:

Ali Raza

Charles Border

Harry Manifavas

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Science in Networking & Systems Administration

B. Thomas Golisano College
Of
Computing and Information Sciences

Rochester Institute of Technology
RIT Dubai

30th May, 2017

Abstract

The last decade has witnessed a major change in the types of traffic scaling the Internet. With the development of real-time applications several challenges were faced within traditional IP networks. Some of these challenges are delay, increased costs faced by the service provider and customer, limited scalability, separate infrastructure costs and high administrative overheads to manage large networks etc. To combat these challenges, researchers have steered towards finding alternate solutions.

Over the recent years, we have seen an introduction of a number of virtualized platforms and solutions being offered in the networking industry. Virtual load balancers, virtual firewalls, virtual routers, virtual intrusion detection and preventions systems are just a few examples within the Network Function Virtualization world! Service Providers are trying to find solutions where they could reduce operational expenses while at the same time meet the growing bandwidth demands of their customers.

The main aim of this thesis is to evaluate the performance of voice, data and video traffic in a virtualized service provider core. Observations are made on how these traffic types perform on congested vs uncongested links and how Quality of Service treats traffic in a virtualized Service Provider Core using Round Trip Time as a performance metric. This thesis also tries to find if resiliency features such as Fast Reroute provide an additional advantage in failover scenarios within virtualized service provider cores.

Juniper Networks vSRX are used to replicate virtual routers in a virtualized service provider core. Twenty-Four tests are carried out to gain a better understanding of how real-time applications and resiliency methods perform in virtualized networks. It is observed that a trade-off exists when introducing QoS on congested primary and secondary label switched paths. What can be observed thru the graphs is having Quality of Service enabled drops more packets however gives us the advantage of lower Round Trip Time for in-profile traffic. On

the hand, having Quality of Service disabled, permits more traffic but leads to bandwidth contention between the three traffic classes leading to higher Round-Trip Times. The true benefit of QoS is seen in traffic congestion scenarios. The test bed built in this thesis, shows us that Fast Reroute does not add a significant benefit to aid in the reduction of packet loss during failover scenarios between primary and secondary paths. However, in certain scenarios fast reroute does seem to reduce packet loss specifically for data traffic. The physical and logical topologies are given below:

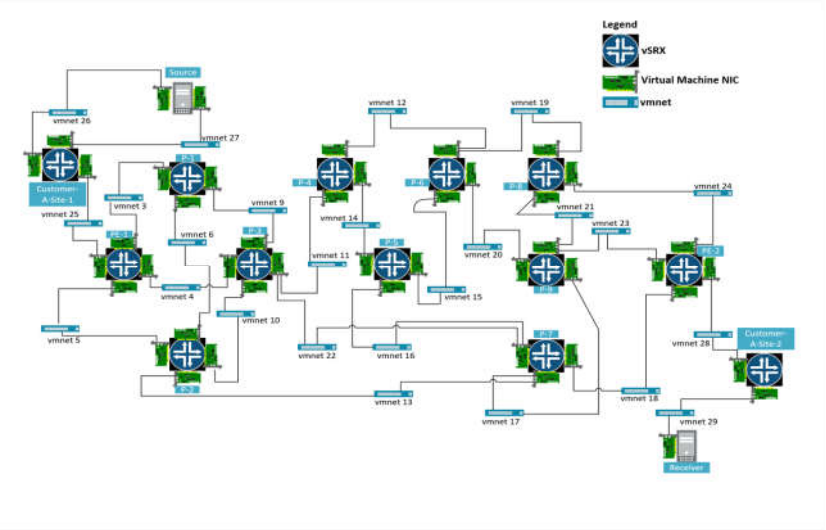

Figure: Physical Topology

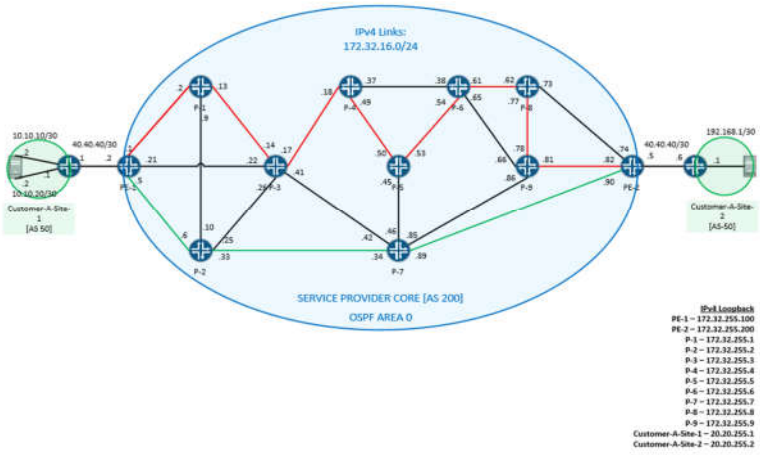

Figure: Logical Topology

Acknowledgements

This thesis is dedicated to Nyshaa and Karsh, the “zero” and “one” that make up my life! I would also like to thank my parents who have always believed in my dreams and given me the courage to overcome all the challenges that lie in its path.

I would also like to thank Dr. Ali Raza, Dr. Charles Border and Dr. Harry Manifavas who have contributed their ideas and helped me see this thesis thru! I have learnt a lot under their guidance and for this I am grateful.

Additionally, I would like to thank Neelay and Ami who made Dubai my home. I would also like to thank Amar and Vrinda under whom I have learnt countless technical skills and life lessons. I would also like to express my gratitude to Derek D’Souza, Varsha Shetye, K-Aunty, Don, Vishwanathan and Murtuza Kukshi who have all contributed to my growth.

1 INTRODUCTION

The last decade has witnessed a major change in the types of traffic scaling the Internet. With the development of real-time applications several challenges were faced within traditional IP networks. Some of these challenges are delay, increased costs faced by the service provider and customer, limited scalability, separate infrastructure costs and high administrative overheads to manage large networks etc. To combat these challenges, researchers have steered towards finding alternate solutions. One of the alternate solutions found were to use Multiprotocol Label Switching (MPLS) in the network. MPLS architectures can be deployed on existing service provider backbones reducing infrastructure costs. MPLS can also be used to interconnect geographically diverse sites while at the same time, reduce the delay found in traditional IP networks.

Over the recent years, we have seen an introduction of a number of virtualized platforms and solutions being offered in the networking industry. Virtual load balancers, virtual firewalls, virtual routers, virtual intrusion detection and preventions systems are just a few examples within the Network Function Virtualization world! Service Providers are trying to find solutions where they could reduce operational expenses while at the same time meet the growing bandwidth demands of their customers. Certain service providers are trying to integrate these virtual platforms into their network, either as virtual Customer Edge device's or route reflectors etc.

A number of advantages are possible with the help of a virtualized core such as reduced infrastructure costs, reduced cooling and power requirements, reduced space requirements in a data center, faster testing and rollout of services, scalability on the fly and simpler manageability of devices. However, limited research has been carried out on the performance of these virtualized networks.

Most service providers have a replication of their existing physical network in a test lab. This tends to increase costs and other overheads (power, cooling, Data Center rent etc.). Certain service providers also make use of commercial software which tries to give them a real-world analysis by simulating their network and its traffic.

The main aim of this thesis is to evaluate the performance of voice, data and video traffic in a virtualized service provider core. Observations are made on how these traffic types perform on congested vs uncongested links and how Quality of Service treats traffic in a virtualized Service Provider Core. The thesis also tries to find if resiliency features such as Fast Reroute provide an additional advantage in failover scenarios.

1.1 Aims and Objective

The main aim of this thesis is to understand the performance of traffic within virtualized Service Provider Networks.

- To understand the behavior of data and real-time (voice and video) tr
- ffic in a virtualized MPLS core.
- To understand in which scenarios can we benefit from Quality of Service in a virtualized MPLS core.
- To understand if resiliency features like Fast Reroute aid in reducing traffic loss during failover scenarios in a virtualized MPLS core.
- Choosing appropriate performance parameters such as Round-Trip-Time and Packet Loss to ascertain the performance of real-time traffic and resiliency within a virtualized service provider core.
- Designing eight varied scenarios on the same network topology to gain a better understanding of the performance of real-time traffic under different scenarios and analyzing the results by comparing the above metrics.

- All results are graphed and observations are made to gain a better understanding.

1.2 Scope of Thesis

This thesis covers the design, configuration and tests required to analyze the performance of real-time traffic in a virtualized service provider core which is relevant to my thesis. More importantly, this thesis tries to gain a better understanding of how Quality of Service affects traffic patterns and if resiliency methods like Fast Reroute in a virtualized environment can prevent packet loss during times of failover between primary and secondary paths in virtualized service provider cores. This thesis does not go into the deep architecture of MPLS, hardware or software optimization to improve the performance of either MPLS or virtualized networks. Details of the encoder-scheme used in VoIP and video traffic captures are also out of the scope of this thesis.

1.3 Research Questions

The following research questions are answered in this thesis.

[1] How does real-time traffic perform when Quality of Service is disabled across congested and uncongested paths in a virtualized MPLS core?

[2] How does real-time traffic perform when Quality of Service is enabled across congested and uncongested paths in a virtualized MPLS core?

[3] Does Quality of Service improve the overall performance on congested and uncongested paths in a virtualized MPLS network?

[4] Does Fast Reroute add significant resiliency in a virtualized MPLS network?

1.4 Contribution

Within this thesis, I use performance metrics such as Packet Loss and Round Trip Time within the Junos RPM suite to comprehensively analyze the performance of Quality of Service and Fast Reroute in virtualized service provider networks. This thesis also explores how real-time traffic performs across eight different network scenarios. In order to validate the eight network models, Juniper Networks virtual SRX (Firefly Perimeter) running on VMware Fusion as the chosen hypervisor is used. Junos RPM is then enabled which helps me track and analyze the performance of real-time applications like voice, video and data in a virtualized service provider core. With respect to the comparative analysis, I found, a trade-off exists when introducing QoS in virtualized networks. When QoS is enabled, we see increased packet drops during periods of high congestion, however we gain an advantage in having lower Round-Trip Times. On uncongested paths, we see similar results when QoS is enabled when compared against QoS disabled scenarios. Based on the testbed created for this thesis, we can observe that a significant advantage is not gained to minimize packet loss during failover scenarios when using Fast Reroute for real-time traffic.

1.5 Related Work

"The rapid growth of the Internet has fueled the development of new technologies that enable IP backbone networks to be engineered efficiently" [1]. I agree with the authors in respect to the above statement. With the growth of Internet traffic in different forms (voice, video & data) along with meeting customer SLA's, service providers are now leveraging an alternate solution in the core such as MPLS, which allow them to tweak the flow of traffic using different traffic engineering techniques, which are not possible using IP only protocols. This inclined me to research more on MPLS.

According to a white paper by AT&T, the authors emphasize that MPLS is here and is not going out anytime soon. They further compare "Traditional" vs "Enhanced Service" architectures**.** Traditional implies an IP only network, typically a VPN run between the customer sites. The authors also mention that a single commercial internet connection doesn't have the capacity to support converged applications. **"**As a remedy, enterprises end up purchasing and managing multiple connections to get the capacity they need during peak traffic periods, which adds cost and complexity to the equation" [2]. A special mention about video traffic is also given which says that video traffic is the fastest growing application generating traffic today and has the ability to knock out other forms of traffic. To counteract the problem, enterprises usually purchase multiple connections sending different types of traffic on each connection. Another issue with traditional VPN's especially in hub-and-spoke IPSec VPN's, is the introduction of jitter and latency which is problematic for real-time communications. Also, multicast isn't supported well when it comes to traditional IP VPN's, as one needs to replicate traffic for each tunnel from the source to the receiver, as IPsec VPN's are generally point-to-point in nature. However, using MPLS we can address the shortfalls found in traditional IPsec VPN's, such as providing Class of Service for prioritizing, managing traffic, reducing delay, jitter and latency. Using MPLS, an inherent flat any-to-any model can be used connecting all sites to increase the performance to exchange voice, video & data traffic. Also, MPLS networks support multicast networks which add to bandwidth savings. At the same time since in most cases there is a single operator for management and control, traffic engineering and end-to-end CoS is simplified.

In an article published in the International Journal of Computer Applications the authors use OPNET Modeler 14.5 to compare results for a MPLS network vs Frame Relay using video conferencing as a traffic load generator for the network. What interests me about this paper is the fact that the authors have used video as a deciding factor. Three major parameters are

compared, i.e. Ethernet delay, end-to-end delay and traffic received. The authors research shows that "Enterprises and service providers can experience an improvement in the rate of achievement of business targets by implementing and maximizing the capabilities of MPLS in their networks. The service providers are suffering from the huge routing table; this can be solved by MPLS because instead of forwarding packets based on IP address we will have router forwarding packets based on labels in these packets. By using MPLS based on labels instead of doing the layer 3 lookups for IP address we can get a lot more performance by using labels" [3]. The graph in Figure 1.1, shows the end-to-end delay for video traffic. It's clearly observed that MPLS performed better, reducing the latency between the source and destination by over six times!

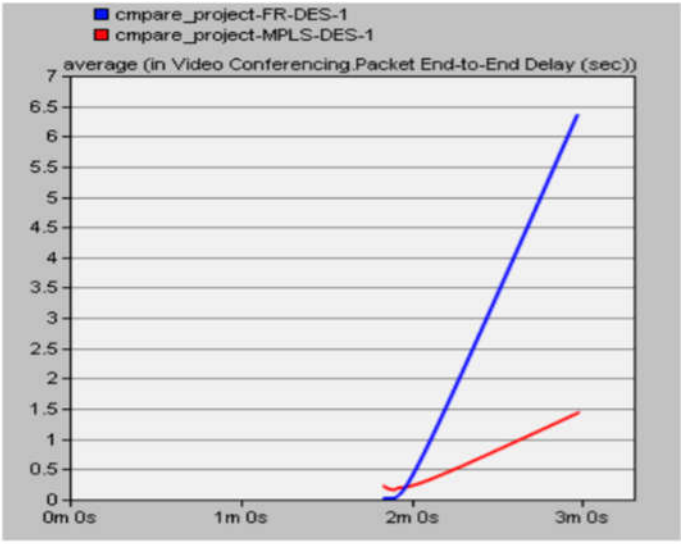

Figure 1.1: End to End delay [3]

"Label distribution in MPLS VPN's, requires the use of an underlying IGP or static routing" [4]. The authors used RIPv2 to understand p2p Queuing delay, VPN delay and LSP delay by introducing VoIP traffic. Their research showed that RIPv2 is not a scalable architecture for MPLS VPN's in large service provider networks due to several factors which were not only limited to long delay's but are also related to the architectural limitation of the protocol's maximum hop counts. The protocol was found to be suitable only in small-

medium Service Provider Networks. This discovery turned my thinking towards understanding the performance using an alternate IGP, such as OSPF or IS-IS.

In separate study conducted, the authors performed an evaluation of Voice Traffic over an MPLS Network using TE and QoS on OPNet Modeler 16.0. Figure 1.2 below, shows the average jitter found when using MPLS TE (without QoS) vs MPLS-TE implementations using different QoS implementation's such as priority queuing, weighted fair queuing and first in first out queuing. The jitter value is plotted against simulation times. It's evident from Figure 2 that WFQ does not show good performance as it is compromising jitter as it is more concerned with congestion avoidance and maintaining the fairness.

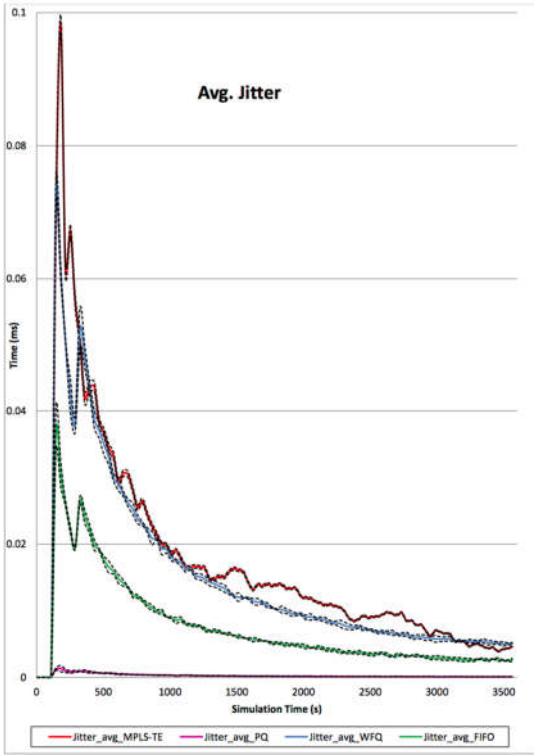

Figure 1.2: Average Jitter on QoS Implemented MPLS-TE networks [5]

The end result of their simulation showed that using TE along with QoS in MPLS network decreased the overall latency, jitter, packet delay variation and end-to-end packet delay as compared to using TE along for voice traffic [5].

A quantitative analysis of labelled (MPLS) vs unlabeled (IP) packets is done on MATLAB Software [6]. Specific amounts of traffic were sent across both the networks (IP vs MPLS), results showed that MPLS has some advantages in comparison to an IP network, especially when it comes to speed. The graph seen in Figure 3 below, shows us results when 1.7 GB was sent from Server 3 towards Server 1, Server 2 and Server 4. It takes more time to reach these servers via IP only than MPLS, citing that MPLS could be used in large enterprises also to gain performance benefits. Also, different factors are identified because of which MPLS is considered as a good technique for TE.

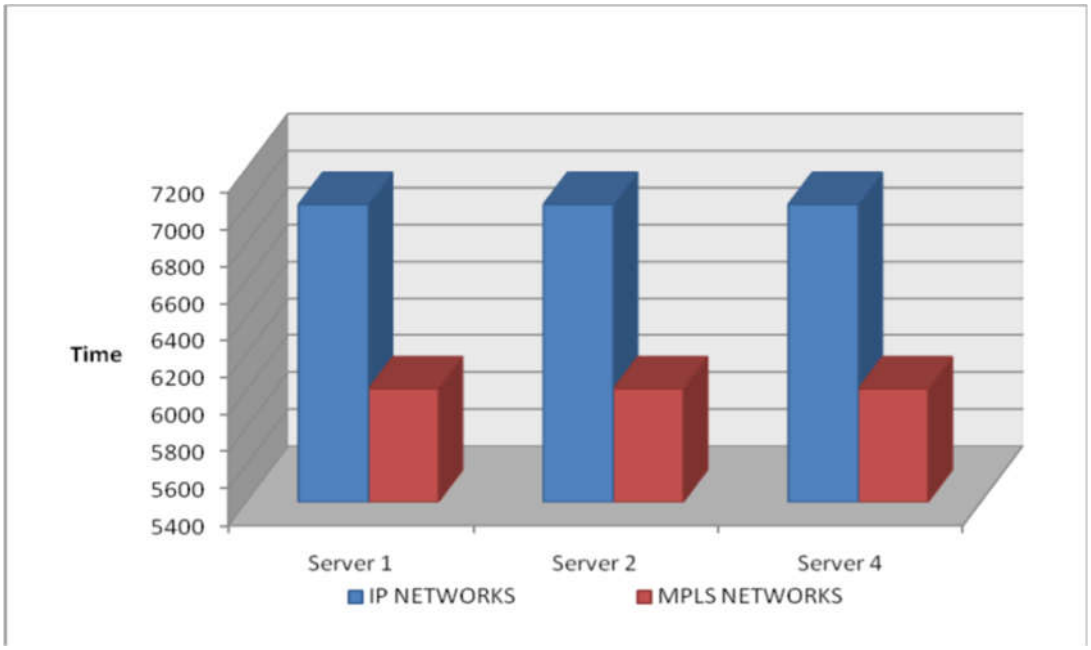

Figure 1.3: Graphical Representation of Server 3 [6]

In [7], showed the power of using MPLS TE. They use the intelligent Constrained Shortest Path First (CSPF) algorithm for MPLS TE to enable QoS routing. Their research showed that using mechanism's and solution's in their paper, service providers can reduce the cost as well as the task of a "complex bandwidth Operation Support System" which associates available queues and allowed bandwidth in the network at the time of provisioning. Overall, the traffic engineered queues leads to a lower overflow possibility for

the tunneled traffic at the same time accelerating the performance for real time traffic such as VoIP.

By comparing the quality of voice traffic with video quality in an MPLS network, the authors showed that prioritizing voice traffic made it perform better as compared to video traffic [8]. In a separate study the authors studied RSVP and CR-LDP using NS-2, a simulation program. After studying the performance, the authors suggested some methods to improve the signaling protocols. “LDP/CR-LDP offers a unified signaling protocol system that provides network operators with the complete label distribution and path setup modes needed for MPLS. The extensions to RSVP provide the capability to establish CR-LSPs with downstream-on-demand label allocation, distribution, and binding only” [9]. In a separate paper written [10] to understand the performance of MPLS networks for VoIP applications using TE signaling and QoS algorithms for 7 different audio codec’s, a conclusion is made that the maximum number of calls are achieved when applying RSVP TE as a signaling protocol along with WFQ as a scheduling algorithm. Analysis showed, to gain the lowest end-to-end delay, packet delay variation and jitter the priority queuing algorithm is best suited.

In a study of IP TV & VoIP Performance in IP, MPLS and ATM Networks, the authors, found that implementing TE alongside MPLS is more suitable for real time applications such as IP TV and VoIP. The major benefit claimed is “MPLS satisfies the condition to force application flows into the path’s which guarantee bandwidth while DiffServ can provide queueing services” [11].

Kathiresan studied the performance of MPLS over IP Networks using Cisco’s IP SLA feature. He observed the performance by building two scenarios on GNS3 to measure the Round-Trip Time, Mean Opinion Score and Latency. His findings showed that “a true IP network suffers from a higher RTT, latency and MOS” [12]. Using MPLS he could achieve

high quality calls and it was found that label lookups were more efficient than routing lookups.

To compare the performance of video multicasting over ATM and MPLS Networks, several tests were performed on OPNet in a paper titled, “Performance comparison of video multicasting over Asynchronous Transfer Mode (ATM) & Multiprotocol Label Switching (MPLS) networks” [13]. The results listed a disadvantage using ATM as the architecture does not support multicast natively and requires the use of LES/BUS server. Even after adding the LES/BUS server MPLS performed much better. They cited a reason of ATM belonging to the Data Link layer whereas MPLS as well as Multicast protocol belong to the Network Layer. Without QoS, MPLS and ATM had a lower throughput as compared to the video source as can be seen in Figure 1.4. The author’s cited the links being shared, with background traffic (ftp and email traffic) along with video source.

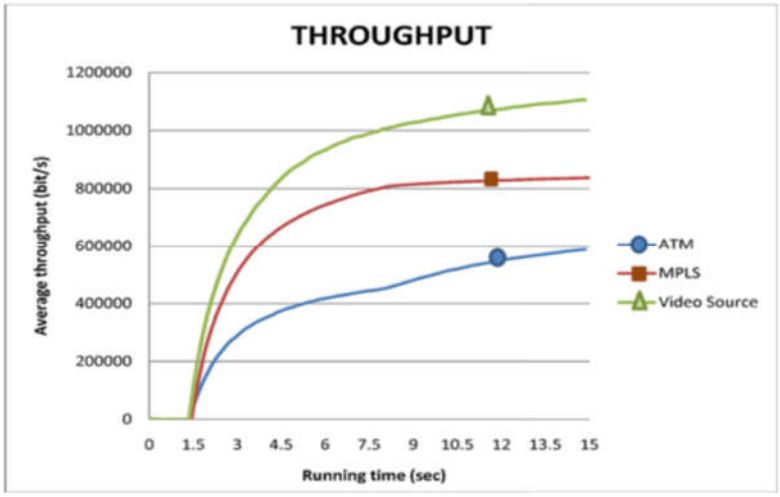

Figure 1.4: Video throughput over MPLS & ATM [13]

The authors [14], use OPNet Modeler to compare six different scenarios to evaluate the performance of Real-Time Applications over DiffServ/MPLS in IPv4/IPv6 using various performance metrics such as end-to-end delay, packet delay variation, packet loss, queuing delay and throughput. It is found that IPv6 experiences more delay and loss than IPv4. It was

also found that the average queuing delay in DiffServ IPv6 networks and in DiffServ/MPLS IPv6 was higher than IPv4 networks. An interesting observation made by the authors is Packet Loss for video traffic was seen higher in IPv4/IPv6 networks, with almost no packet loss seen in DiffServ/MPLS IPv4/IPv6 networks.

In a paper published by International Journal of Research in Computer and Communication Technology, the authors use GNS3, a graphical network simulator and Cisco IOS images to demonstrate feature's such as Fast Re-route and Node Link Protection in a paper titled, "Traffic Protection Against Link And Node Failures Using Fast Re-Route For MPLS Networks" [15]. However, I feel though this paper aimed to give an understanding of the technologies, comparisons against MPLS networks lacking traffic protection wasn't clear. Also, no distinction was made with respect to the performance of video traffic using the above mechanism. Another drawback of the approach mentioned in the paper is the emulator is only suitable for Lab environment's and cannot be used in production environments.

According to a paper published in International Journal of Innovative Research in Computer and Communication Engineering, the authors ran simulations tests in NS2 to develop efficient QoS Routing Algorithms for Protection of Data Flow in MPLS Networks. Their research showed that as the number of faults increased so did the time for recovery and fault detection increase as seen in Figure 1.5. Authors used Packet Delivery Ratios, Throughput and Fault Recovery Time as tools for performance measurement against their own algorithm. Their research finally concludes with "emphasizing on the use of fast recovery techniques in MPLS to guarantee QoS in todays networks" [16].

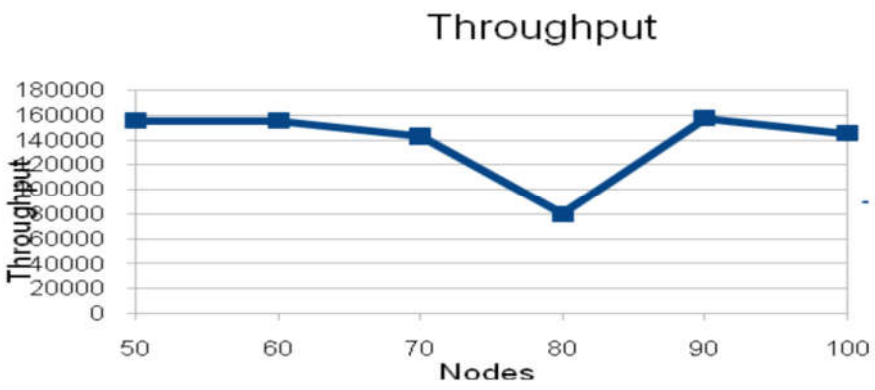

Figure 1.5: Throughput versus number of Nodes [16]

1.6 Comparison of Related Work

From my literature review its evident that MPLS offers several benefits and paves the way for future network scalability and performance. However, during my literature review I felt that a lot of research has been done to compare traditional IP networks against MPLS networks either based on theoretical analysis or analytical models. Lesser research has been carried out on real-time traffic. Although a few papers did analyze the performance of video traffic, majority research is conducted comparing only voice and data traffic.

Another significant point to mention is past research has been carried out only on network simulators like OPNet Modeler, GNS3 and NS2. Although this method is beneficial to ascertain results, it cannot actually replicate the commercial grade virtualized platforms (which eventually are used in production networks) used in this thesis. I also found that although research conducted to understand the performance of MPLS, there was limited research made to understand the performance of resiliency features within the MPLS portfolio, especially on virtualized networks.

In this thesis, my work focuses on using virtualized software such as Juniper Network's Firefly Perimeter (vSRX) to understand the performance of MPLS on real-time

traffic in a virtualized service provider core. I also wanted to understand how Quality of Service affects the above traffic types while at the same time studying the applicability and performance benefits, if any from resiliency feature such as Fast Reroute in MPLS. Round-Trip Time has been used to help as a performance parameter. The results are plotted on a graph following which observations are made.

1.7 Thesis Outline

This section describes the outline of the thesis in brief.

- Chapter 2 provides a brief background of the protocols, software and technologies used thru this thesis.
- Chapter 3 provides the readers with information on how the test bed is built as well as how the results are ascertained. It speaks in depth about the hardware and software used as well the traffic flow for all the scenarios. This chapter also speaks about the challenges faced during the thesis.
- Chapter 4 explains eight different scenarios and the twenty-four tests carried out to understand the performance of real-time traffic in a virtualized service provider core.
- Chapter 5 concludes this thesis and answers the research questions in Chapter 1 of this thesis.
- Chapter 6 speaks about the future direction and additional research required with respect to this thesis.
- Chapter 7 provides the references for the articles cited thru this thesis.
- Chapter 8 includes the appendix section sharing the configurations for all the virtual routers.

2 BACKGROUND

2.1 Internet Protocol

The original implementation of Internet Protocol was meant to be used within an interconnected system of packet switched networks where a protocol was needed to move traffic between the source and destination in a network. Internet protocol was developed as part of the initial DARPA Internet Program. Earlier to the Internet, was a super network created within the Advanced Research Projects Agency (ARPA) in 1969. The main aim of this super network was to design a fault-tolerant network that would enable the United States government to function even in times of a nuclear attack. As time passed this program paved the way of extending the concept of networks beyond military use. Today the Internet Protocol fuels the Internet. This protocol has several advantages, not limited to being open standard, interoperable, reliable (when paired with Transmission Control Protocol) and simple addressing scheme [17].

This protocol provides a mean of sending blocks of data within a datagram to hosts within an IP Network. These hosts are further identified with the help of logical addressing provided by Internet Protocol. Logical addressing is provided with the help of separating a host and a network with the help of a "subnet mask" [18]. This protocol has two simple functions apart from moving traffic. These functions are fragmentation and addressing. The Internet Protocol runs within the TCP/IP stack at the Internet layer. Two versions of Internet Protocol are currently deployed, Internet Protocol version 4 and version 6. For the scope of this thesis, Internet Protocol version 4 is explained below.

2.1.1 Internet Protocol version 4 (IPv4)

IP's functions were initially integrated with TCP. Over time this functionality was separated from the three earlier versions of TCP and thus coined the term Internet Protocol version 4 (IPv4) [19]. Figure 2.1, gives a brief overview of the Internet Datagram Header.

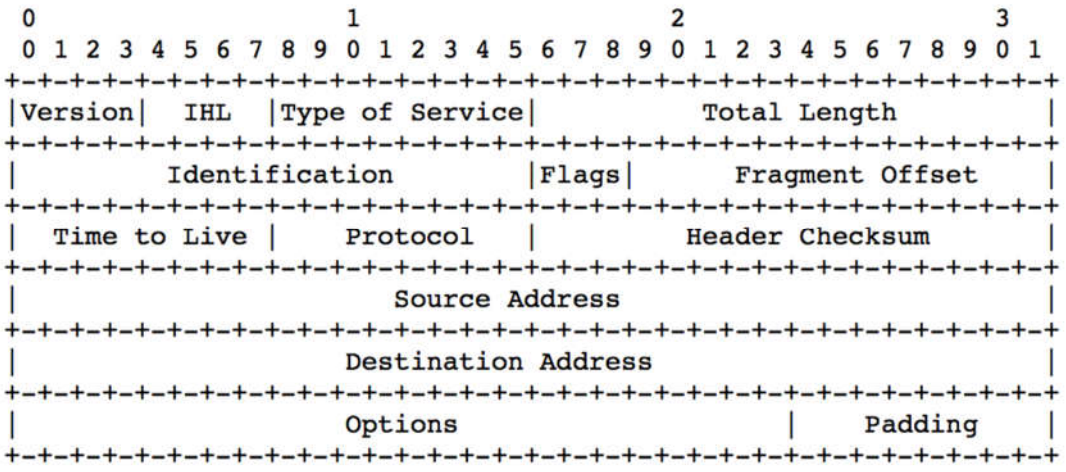

Figure 2.1: Internet Datagram Header [19]

- Version: This field speaks about the format of the internet header being used. IPv4 uses version 4 in this field.
- IHL: This field speaks about the length of the internet header depicted in 32 bit words. This field also shows the beginning of the data.
- Type of Service: This 8-bit field is used as an indicator on how traffic within the datagram must be treated e.g. Higher precedence traffic over lower precedence traffic. This is useful for creating a traffic classification between network control, voice, video and data traffic.
- Total Length: This field included the total length of the header and data portion of the datagram in octets.

- Identification: This field is assigned by the sender and helps in assembling of fragmented datagrams.
- Flags: This field is used to show if more fragments are expected or if it's the last fragmented internet datagram.
- Fragment Offset: This field is used to indicate the position of the fragment in the datagram.
- Time to Live: This field is decremented across next-hop routers and shows for how long the packet can stay alive in the internet system.
- Protocol: This field is used to show the next level protocol used in the data portion of the internet datagram.
- Header Checksum: This is a checksum calculated at every hop to make sure of the headers integrity.
- Source Address: This 32-bit field represents the source's logical IP address.
- Destination Address: This 32-bit field represents the destinations logical IP address.
- Optional: This field is optional in a datagram.
- Padding: This variable field is used to make sure the internet header ends on a 32-bit boundary.

2.2 Multiprotocol Label Switching (MPLS)

Conventional IP forwarding requires each router to evaluate every packet against a "Routing Table". Within this table, the routers, routing process perform a longest match lookup to find the egress interface and next hop for the packet based on the destination address by analyzing a packets header. Although this process enables us to forward packets in traditional IP networks, the process tends to be slower and more resource consuming.

In 2001, a vendor neutral technology was developed which could accelerate the forwarding process within routers as well as lead to several other advantages. Multiprotocol Label Switching is the name given to this technology. Some of the benefits of MPLS are [20]:

1) Scalability of Network Layer Routing: With MPLS labels we can aggregate forwarding information whilst working in presence of routing hierarchies. An example of this is a Layer 3 VPN, where only edge (PE) routers maintain customer specific routes whilst the provider/core routers are VPN-unaware routers. These core routers only forward traffic based on the labels.
2) Greater Flexibility in delivering routing services and traffic engineering: Using MPLS labels we can bind traffic flows to either specific paths in the network or even bind Quality of Service parameters to make sure this traffic is treated in a unified manner within the core. Traditional IP routing lacks native traffic engineering properties as all traffic is forced to use a path even though multiple equal-cost paths may be available. Changing the cost/preference of links to alternate paths would in turn leave one link over utilized and other links underutilized within traditional IP networks (Non-MPLS). However, using MPLS we can traffic engineer paths by creating label-switched-paths across different link and redirecting flows to different label-switched paths.
3) Increased Performance, Resiliency and Scalability: By using label-switching a router can reduce the overhead of performing longest match lookups thru a routing table. Apart from the benefit of reduced overhead on the router, MPLS offers features such as fast reroute which provide an additional layer of resiliency and scalability in the network.

4) Reduced Infrastructure Costs: Another advantage of MPLS lies in its reduced infrastructure costs. The reason for this is MPLS can run on virtually all common underlying technologies such as Ethernet, ATM, Frame Relay.

2.2.1 MPLS Header

When a router enters the MPLS label switched path, the ingress router adds an MPLS Header to the packet via a "push" operation. This MPLS Header is finally removed via a "pop" operation either by the Pen-Ultimate Hop Router or by the egress router. Figure 2.1 shows how the MPLS header looks. The MPLS header is a 32-Bit MPLS Shim Header which is added between the L2 Header and Data Payload. The MPLS header consists of the following [21]:

- Label: A 20-bit Label identifies the packet to a label-switched-path. This value changes across hops on a label-switched-routers.
- Class of service (CoS) (experimental bits): Originally this field was designed for Class of Service, however it did not receive any formal usage and hence has been coined as experimental bits. In production networks, these bits are used as Class of Service bits to treat traffic in a specific manner across the label switched path.
- Bottom of stack bit (S): This field is used to indicate if the MPLS packet has more than one label associated with it. If a value of 1 remains, the label switched router knows that after popping this label only an unlabeled packet will remain.
- Time to Live (TTL): This field is used to show how many hops can the MPLS packet transverse. It is decremented along each hop in the MPLS label switched path and the packet is discarded if the value drops below 1.

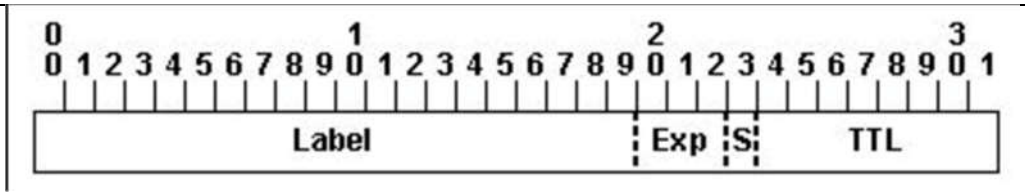

Figure 2.2: MPLS Header [21]

2.2.2 MPLS Elements:

An MPLS domain consists of a group of MPLS enabled routers forwarding MPLS packets across predetermined paths from one MPLS-enabled router to another MPLS-enabled router [22]. Main terminologies are discussed below:

- Label Switched Router: A router which is enabled for MPLS operations and forwards packets based on the MPLS label is called a Label Switched Router. This router typically checks a label forwarding information base (LFIB) and performs push, swap, pop operations.
- Label Edge Router: There are two types of label edge routers. When a packet enters the MPLS domain, it meets the Ingress Label Edge Router. This router typically performs the “push” operation. When the packet leaves the MPLS domain it goes thru the Egress Label Edge Router who may perform a pop operation and an additional lookup based on Layer 3 information in certain scenarios.
- Forward Equivalence Class: A group or set of packets which has similar characteristics and are being forwarded with the same priority along the same path will all be bounded with the same MPLS label. This mapping is called a “Forward Equivalence Class (FEC)”. Every packet is assigned with a FEC only once at the ingress router.
- Transit Router: Routers which are typically VPN-unaware routers and make forwarding decisions solely based on the MPLS Label are termed as transit routers.

- Penultimate Router: The second last or Pen-Ultimate router in the label switched path can sometimes (based on configuration) perform a "pop operation." This is done for reducing the load on the egress router to perform a MPLS label pop and a route lookup. This router is called a Pen-Ultimate router.

2.2.3 MPLS Architecture

The MPLS architecture is split into two separate planes i.e. the control plane and the forwarding (data) plane. Every MPLS enabled node would require these two planes for building its label information base and passing MPLS traffic within the MPLS domain.

In the control plane the IP routing protocol (RIP, OSPF, IS-IS) exchanges routing information and builds the IP routing table. MPLS protocols such as LDP or RSVP make use of the routing table and exchange label bindings between MPLS routers. The label bindings are then pushed to the Label Forwarding Table within the Forwarding Plane of the MPLS node. The Label Forwarding Table is used referenced for MPLS packet lookups [23]. Figure 2.3 shows an example of the Basic Architecture of an MPLS Node.

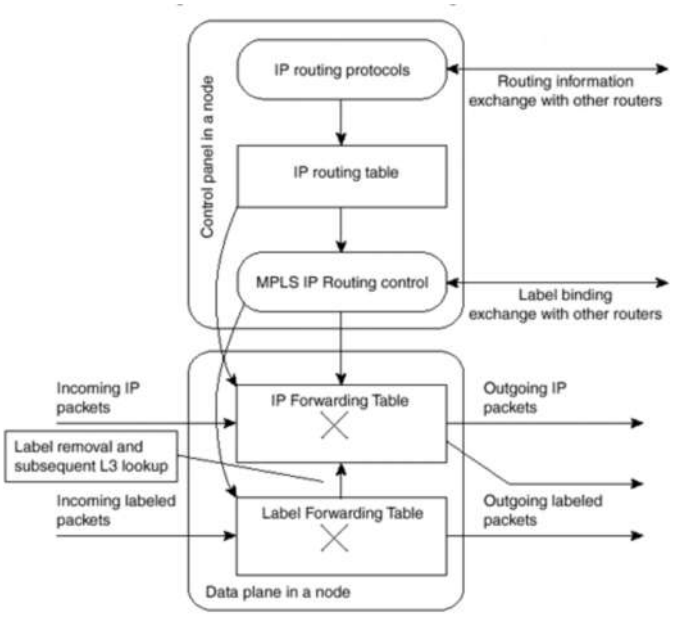

Figure 2.3: MPLS Architecture [23]

2.2.4 Label-Switched Path

A label switched path is a unidirectional path through which packets are sent from the ingress to egress node. This path can be statically configured or can be provisioned with the help of MPLS protocols like RSVP and LDP.

2.2.5 Explicit Route Object (ERO)

Using an Explicit Route object, the MPLS domain administrator can explicitly configure a set of strict or loose next-hops. These next-hops would dictate which label-switched-routers the label switched path would have to traverse over. A “Strict” next-hop implies that the next-hop should be the same as in the Explicit Route Object List. A “Loose” next-hop implies that somewhere in the path to the egress node, the label switched path must transverse the “loose” next hop in the Explicit Route Object List. This feature is generally configured on the ingress MPLS router.

2.2.6 Bidirectional Forward Detection for MPLS LSPs

Certain situations may arise which leads to black holing of user data traffic within an MPLS network. An example of this situation is where the control plane on an MPLS LSP is functioning however one of the label-switched routers are having a fault in the forwarding plane. This would lead to black holing of user data traffic [24]. BFD for MPLS provides a mechanism by which the Ingress Label Switched Router sends periodic light-weight BFD hellos to the egress node. A reply to the hello before the receive interval expires informs the ingress label switched router the label switched path is functioning correctly.

2.3 Resource Reservation Protocol (RSVP)

RSVP was initially a host-to-host protocol used to reserve resources among hosts. The functionality of RSVP was further extended to make it a router-to-router resource reservation

protocol. “RSVP requests resources for simplex flows, i.e., it requests resources in only one direction [25].” Thus, RSVP treats the sender and receiver as separate logical entities, however in certain scenarios the process may act as a sender and receiver at the same time. RSVP is not a routing protocol, instead it piggy backs on IPv4 or IPv6 as a transport protocol.

RSVP flows are unidirectional in nature. RSVP has several benefits apart from MPLS Label distribution which are not found in LDP. The major benefit received thru RSVP is Traffic Engineering where the ingress node can signal bandwidth, MTU or Explicit Route Objects as parameters when signaling the MPLS LSP. This allows the MPLS domain administrator to signal MPLS LSPs on paths other than the IGP shortest path.

RSVP also makes use of the Constrained Shortest Path First (CSPF) algorithm to calculate label switched paths passed on constraints specified by the MPLS domain administrator. LSP attributes such as Link Coloring/Affinity/Administrative Groups, bandwidth requirements and Explicit Route Objects or Link Attributes such as colors on specific links and available bandwidth could be used by this algorithm to choose on which paths the MPLS LSP should be signaled [26].

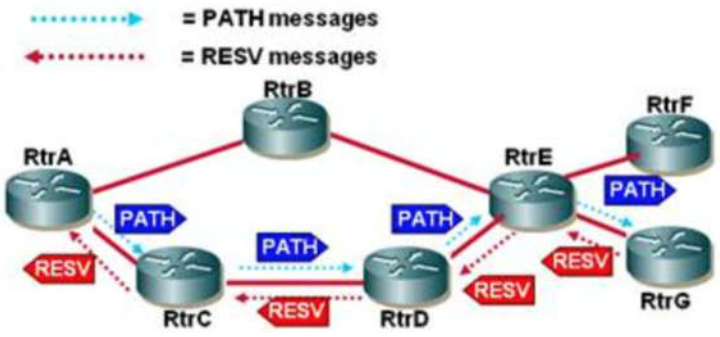

Figure 2.4: RSVP Signalling [63]

Figure 2.4, illustrates the signaling mechanism in RSVP. RtrA wants to establish an MPLS LSP signaled thru RSVP with RtrG. RtrA initiates a “PATH” message downstream. The path

taken by the PATH message could be the underlying IGP's path or alternatively a specific set of hops could be taken if specified within an ERO by RtrA. Once the "PATH" message reaches RtrG, RtrG responds with a "RESV" message upstream towards the ingress node, RtrA. The reservation message is sent by every node in the path to the upstream router informing which label to use when sending traffic within the label switched path. When RtrA receives the "RESV" message from RtrC, he knows that nodes along the path have adequate resources to pass traffic and creates a label switched path along with the label signaled by RtrC locally.

2.3.1 Link Protection

The link protection mechanism in RSVP is a feature used for resiliency in MPLS networks. The main purpose of this feature is to ensure traffic destined for an egress node continues to reach it even if there is a failure on the local nodes interface carrying traffic towards the egress node. This is done with the help of creating a bypass LSP which would handle traffic if the primary interface were to fail. The bypass LSP can be statically configured or be provisioned dynamically with the help of the CSPF algorithm. The bypass LSP cannot have the same egress interface as the LSP being monitored. If for any reason the link-protected interface were to fail, traffic would be quickly switched to the bypass LSP [27].

2.3.2 Fast Reroute

The Fast Reroute feature is a feature used to reduce packet loss in an MPLS network. This feature is signaled by the ingress node by adding an object to the RSVP PATH message requesting downstream nodes to create a detour point. The intermediate label switched routers then originate detour PATH messages, to detour the label switched path around the label switched routers downstream link and node. These paths are maintained in the memory

of every intermediate label switched router. The advantage of this approach is, in case of failure of the primary path the detour path is always present on the intermediate router. The intermediate router would then detour the traffic over the detour path and signal the ingress node of a link failure by sending a "PATH ERR" message. The advantage received here is traffic is not black holed. The ingress router then has the option to use an alternate path if available to reach the egress router [28].

2.4 Quality of Service

One of the biggest challenges facing todays Internet infrastructure is Quality of Service. Quality of service, popularly called QoS revolves around applications and has evolved since its original implementation due to the rapid growth of high bandwidth and low latency applications. The initial implementation of the internet dwelled around data oriented applications where Quality of Service was not the prime criteria. Over the years with boom of the World Wide Web, voice and video applications were integrated and this led to several challenges as both these traffics required special treatment within a network for optimal performance. Quality of Service can be observed in many forms depending on which network we focuses on as well as which application QoS revolves around. To further illustrate this point, QoS could be used in service providers and enterprise networks to increase throughput/bandwidth, minimize latency and congestion in the network or control bursts in the network. On the other hand, QoS could revolve around voice or video applications as well by trying to reduce jitter, packet loss, delay and increase overall throughput. The key principle of QoS is to guarantee service and support as a framework to the Internet [29] [30].

2.4.1 Quality of Service Architectures

The IETF has made significant efforts in this area along with several other researchers. A brief background of the three important architectures around QoS are explained below [31]:

2.4.1.1 Integrated Services (IntServ)

The IntServ Architecture was the first architecture proposed by the IETF. This architecture was based on making reservation across routers over the Internet on a per flow basis. These reservations would be kept in soft state and would need to be periodically refreshed. Resource Reservation Protocol (RSVP) would be used to signal requirements such as bandwidth to make a reservation. However, since this method required frequent refreshes it could not scale well in large networks such as the Internet and thus the IntServ model was not widely deployed.

2.4.1.2 Differentiated Services (DiffServ)

Unlike IntServ, the main goal of DiffServ was to provide coarse-grained control and provide scalability. This was done by aggregating multiple traffic flows into one. Thus, not needing to allocate bandwidth to individual flows. Another key difference is IntServ follows a more dynamic approach of reservation with the help of RSVP, whilst DiffServ needs to be configured statically on every hop in the network. DiffServ makes use of packet classification either based on fixed fields (e.g. Source/Destination IP, Port Numbers) or Type of Service bits and pushes the traffic inside Queues. Within the queues, scheduling takes place and finally packets are transmitted out of the router. “DiffServ is widely used in the Internet, with or without MPLS [30].” An important component within this architecture is PHB (Per Hop Behavior). The concept of PHB explains how a DiffServ node would handle traffic belonging to a specific forwarding class.

2.4.1.3 Multiprotocol Label Switching (MPLS)

MPLS is used in IP Networks along with DiffServ to provide Quality of Service. The combination of these two technologies enable us to guarantee resources for real-time services such as voice and video. MPLS achieves this by creating a connection prior to the actual transmission of data packets. This connection reserves resources along the path by making use of Resource Reservation Protocol. Traffic can only pass between the sender and receiver once this logical path comes up. Post transfer, this connection is generally torn down and the bandwidth reservation are freed. Label Switched Path can be set up across explicit paths further reducing the end-to-end delay. Packet classification, policing and queuing services on a per port basis can be provided by MPLS when using in conjunction with DiffServ.

2.4.2 Elements of Quality of Service Architecture:

The following section gives a brief overview of the elements of a QoS architecture:

2.4.2.1 Traffic Classification

Traffic classification implies examining the incoming traffic and associating it with a form of service level. Once classified this traffic is assigned to an output queue. The number of queues supported is typically device dependent. A queue is generally mapped to a "Forwarding Class". Also during classification, traffic gets assigned a packet loss priority value. The packet loss priority value speaks about a packets drop-eligibility during periods of congestion. Four loss priorities can be assigned. These are low, medium-low, medium-high and high. Two common options of classification are mentioned below:

- Multifield Classifiers: A multifield classified is typically applied at ingress when traffic is coming from an untrusted domain, i.e. a domain from where QoS markings

cannot be trusted. Multifield Classifiers make use of one or more fixed fields such as Source IP/Port, Destination IP/Port, MAC Address etc.

- Behavior Aggregate: Once traffic is classified at the edge or marked by a trusted entity, we can use Behavior Aggregates as a means of classification. Behavior Aggregates popularly called "BA" makes use of Quality of Service markings. The advantage received thru Behavior Aggregates is a reduced overhead on the device as BA requires reduced packet analysis. This feature makes BA useful in the core network where high volumes of traffic needs to move quickly. Three types of Behavior Aggregates within the scope of this thesis are explained below:
    - IP Precedence: The original implementation of the IP header had reserved 8 bits as Type of Service (ToS) for Quality of Service. However, in practice only the first three bits, popularly called the "Most Significant Bits" were used. IP Precedence classifies traffic by looking at the first three bits in the ToS field. By using the first three bits, each packet could be assigned a priority as well as request to be treated as either low latency, high throughput or high reliability.
    - DSCP: Instead of using only the upper 3 bits with the ToS field of an IP header, the upper 6 bits are used [31]. The field was then called "DS" (Differentiated Service). Classification based on the requirements of each service of traffic was based on the first six bits. Within the DiffServ PHBs four PHBs have been standardized. They are:
        - Expedited Forwarding: This PHB is designed for traffic such as voice which would require low latency, low delay and low jitter.
        - Assured Forwarding: Within this PHB no delay related parameters are defined. This PHB is designed to control packet loss and supports four

classes (AF1, AF2, AF3, AF4) with each class having three drop probabilities.

- Class Selector Code Points: The CS code points are designed for backward compatibility with the IP-Precedence field. Real world use cases revolve around using these code points for network control QoS.
- Best Effort: Traffic that cannot be classified into the above classes generally falls into the Best Effort class.

2.4.2.2 Policing

A policer essentially forms the first stage of pre-emptive congestion management. Policing lets the network administrator condition the traffic by placing bandwidth and bandwidth burst constraints. By using policers, network administrators can enforce service level agreements. Policing can be done at both ingress and egress levels. Policing is also referred to as rate-limiting.

2.4.2.3 Scheduling

The scheduling phase defines how queues treat traffic. The following parameters can be used for scheduling queues:

- Transmission Rate: This parameter is used to control the amount of bandwidth that can be allotted to an interface.
- Queue Priority: This parameter is used to define the relative importance of the queue when being compared with other queues. A higher priority (in-profile) queue can send traffic prior to a lower priority (in-profile) queue.
- Delay Buffers: This parameter is used to configure the extra amount of data which the queue can use in times of queue congestion. A larger delay buffer supports a larger latency.

- Drop Profile and Drop Profile Maps: A drop profile is used to determine how traffic should be dropped if congestion occurs. This typically occurs when the delay buffer starts getting full. Drop profiles typically check the Packet Loss Priority Value to make decisions on which packets to drop. Drop Profile Maps are used to tie the Drop Profile with a scheduler.

2.4.2.4 Rewrite Makers

By default, when traffic egresses a device, the ToS bits are overwritten. Thus a packet which is classified as Expedited Forwarding at the ingress would be sent out as Best Effort. In order to preserve the correct markings a rewrite marker is configured to correct the Quality of Service bits. Setting correct rewrite bits is beneficial as it reduces the overhead on downstream nodes to classify traffic based on BA markings rather than using Multifield Classifiers.

2.5 Open Shortest Path First version 2 (OSPFv2)

Open Shortest Path First popularly called OSPF is routing protocol used within an autonomous system. An autonomous system is a group or collection networking devices which come under a common administration. OSPF falls under the category of link-state protocols within the family of Interior Gateway Protocols (IGPs). OSPF routers exchange link-state advertisements (LSAs) which contain information about their attached networks and links with OSPF neighbor routers. For two routers to form adjacency certain parameters much match in the “Hello Packet”. These parameters are Network Mask (on broadcast links), hello interval, dead interval and options field. OSPF routers use “Hello Packets” during adjacency formation and also as a keep alive mechanism. A reliable mechanism is used for the LSA flooding. These LSAs are then stored by every OSPF router within a Link State

Database (LSDB). All routers within the OSPF area must have the same link state entries in the link state database for OSPF to function correctly. The OSPF router uses the information found in the LSDB as inputs for the Shortest Path First algorithm to calculate the shortest path to reach the destination network. As per the RFC, a multi-area OSPF implementation must use "Area 0" as the backbone area. This means every area requires to be connected to Area 0, for OSPF to function correctly. OSPF tries to form an adjacency with all neighbors on all interfaces. This tends to create a problem on broadcast medias such as Ethernet as this leads to additional adjacencies being formed over a common link with all routers advertising the same set of information. In order to reduce control traffic on broadcast segments, OSPF makes use of a Designated Router and a Backup Designated Router. These two routers form "Full" Adjacency with each other while "Designated Router Other" form only "2-Way" state. Point-to-Point links such as SONET, do not require the concept of DR and BDR as there are exactly two neighbors involved, thus saving time in getting the adjacency in Full State [32].

2.5.1 Open Shortest Path First Support for Traffic Engineering

By default, MPLS makes use of the IGPs shortest path. At time the shortest path might not be the idle path due to congestion. Traffic Engineering in OSPF allows the network administrator to bypass the standard routing table by moving traffic flows to alternate paths in the network. This is done by enabling traffic engineering for OSPF on all routers within an area. After enabling traffic engineering for OSPF, the routers start generating and exchanging Traffic Engineering information in opaque link-state advertisements. These new LSAs carry traffic engineering information which helps the routers build a new database used solely for traffic engineering computation called the "Traffic Engineering Database (TED)."

Constrained Shortest Path First (CSPF) algorithm then runs on this database to compute paths needed by the MPLS LSP [33].

2.5.2 Node-Link Protection

To add IP fast-reroute capability to OSPF we can make use loop-free alternate paths. A loop free alternate path is used when we do not want to send traffic back through the routing device to reach the destination. A loop-free alternate path depends strongly on the network design as not all destination are reachable thru alternate paths. The node-link protection feature tries to establish a path by using a routing device not in the path of the routers original shortest path [34].

2.6 Bidirectional Forward Detection (BFD)

Certain network devices make use of separate control and forwarding planes. Whilst several protocols exist to track the stability of the control plane, a protocol was needed to detect failures at the forwarding plane level. Bidirectional Forward Detection (BFD) was a protocol developed for this specific purpose. BFD runs as a point-to-point unicast protocol which piggy backs on any data protocol operating at the link-layer, network layer or in tunnels. BFD helps us detect failures between the communication path between two nodes. BFD could be used to track connections across single hop, multiple hop as well as virtual circuits or tunnels. At the back-end BFD uses the concept of three-way handshakes whilst establishing BFD sessions and tearing them down to ensure both the nodes are aware of state changes [35]. BFD works by exchanging BFD packets with a peer and if any of the peers fail to receive a BFD packet within a predetermined time, the BFD process signals the protocol of a problem in the communications path.

2.7 Border Gateway Protocol 4 (BGP-4)

Protocols like RIP, IS-IS, OSPF are used within autonomous systems to exchange routing information. These protocols cannot scale well across autonomous systems. A routing protocol was needed which could exchange routing information between autonomous systems. This led to the development of Border Gateway Protocol popularly called BGP. The main aim of BGP is to exchange routing information with other autonomous systems ultimately building a network of Autonomous systems and pruning looped paths.

BGP also supports Classless Inter-Domain Routing as well exchange of aggregated routes with peer autonomous systems. The benefit received via aggregating routes and exchanging the aggregated routes with peer BGP autonomous systems is scalability. Exchanging aggregated routes over the internet allows us to reduce the size of the global routing table. A router which implements BGP is termed as a BGP speaker. A BGP speaker can be configured for two types of connections, i.e. Internal BGP (iBGP) or External BGP (eBGP). Internal BGP is used in scenarios which require exchanging BGP prefixes and routing information within the same autonomous system. On the other hand, External BGP is used in scenarios which require exchanging network layer reachability information across different autonomous systems.

BGP stores routes in a Routing Information Base (RIB). When a BGP speaker receives BGP routes from his BGP peers, these routes are initially stored in an Adj-RIB-In table. Optionally, policies could be used to filter routes as they are sent to the RIB-Local table. Active BGP routes are then passed to the Adj-RIB-Out table [36]. Routes present in the Adj-RIB-Out table are propagated to BGP peers. Policies can be used to further filter which routes should be prevented from being advertised.

BGP also exchanges a set of route attributes which are broken into four different categories. These categories are:

- Well-known mandatory: These attributes must be present in every BGP update message and be supported in all BGP implementations. Examples of well-known mandatory attributes are AS Path, Origin and Next-Hop.
- Well-known discretionary: These attributes do not have to be included in every BGP updates message, but must be supported in all BGP implementations. Examples of well-known discretionary attributes are Local-Preference, Atomic Aggregator.
- Optional transitive: These attributes do not need to be supported in every BGP implementation but if they are present, they must be sent to other BGP peers, unchanged. Examples of Optional transitive attributes are community and aggregator.
- Optional non-transitive: These attributes do not need to be supported in every BGP implementation. However, if a BGP speaker receives an optional non-transitive attribute and does not recognize it, he does not need to propagate this to his peers. Examples of Optional non-transitive attributes are cluster list and Originator ID.

BGP's main aim is not to find the "shortest path" but to find the "best path". BGP attributes help a BGP speaker decide which routes to consider and which paths to take.

2.7.1 Multiprotocol Extension for BGP-4 and VPN-IPv4 Address Family

The multiprotocol BGP is an extension of the Border Gateway Protocol which enables BGP to carry routing information for multiple network layers and address families [37]. MP-BGP is useful for BGP based deployments of L3VPN, L2VPN, VPLS and EVPNs.

The VPN-IPv4 address family uses MP-BGP to exchange labeled VPN routes between Provider Edge devices. Figure 2.5, shows the structure of the VPN-IPv4 NLRI:

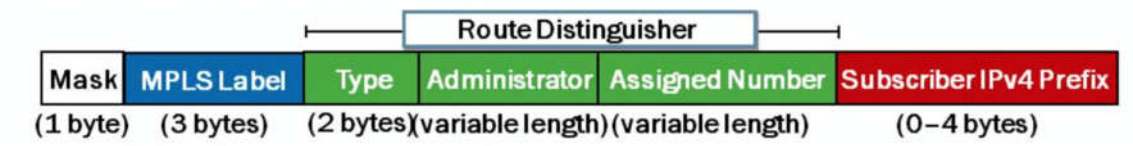

Figure 2.5: VPN-IPv4 NLRI [38]

The VPN-IPv4 NLRI consists of a 1 Byte Mask field. The MPLS label is also popularly called the VRF label as the receiving PE router associates routes for a particular VRF instance with this label. The VPN-IPv4 NLRI also has the route distinguisher present which helps alleviate the problem of overlapping address spaces between customers. Lastly, the subscriber IPv4 Prefix is also included in this NLRI. MP-BGP helps exchange VPN-IPv4 address family routing information between Provider Edge devices [38].

2.8 VMware Fusion 7

VMware Fusion 7 is an application for Intel based Mac OS X users which allows users to create, open and run VMware virtual machines on a Mac. This platform lets Mac users run Microsoft Windows, L inux, x86 operating systems and many other operating systems on the Mac without requiring a system reboot. Due to the virtualization capability, multiple operating systems can run at the same time based on the Mac's hardware. VMware fusion does this by mapping hardware resources of the Mac to the virtual machines (guest os) resources. This gives every virtual machine its own memory, process, I/O devices and storage [39]. A virtual machine when run inside VMware Fusion is also popularly called a guest os. A virtual machine is a software which emulates a physical computer and the operating system and applications running inside it. Elements such as display, storage, processor, memory, disk drives, network adapters and USB controllers are created by software and stored as files on the Mac. [40]

2.9 Junos OS

Junos OS is an operating system built by Juniper Networks running on the principles of reliability, security and flexibility. Junos OS is the operating system which powers network device from Juniper Network's broad device portfolio. Also, tightly integrated with the Junos

operating system is network automation. Junos OS is a robust and modular operating system built on Free BSD Unix operating system. Juniper Networks has also removed services not pertaining to the devices functionality, thus security hardening the device. A few important features of Junos OS are [41]:

- Modular Software architecture: Each process in the Junos OS is allocated its own dedicated memory space. Thus, if a single process fails the entire device does not need to be restarted. An individual only needs to restart the failed process.
- Ease of Management thru Configuration Hierarchies: Junos OS breaks the configuration into meaningful hierarchies for simpler manageability. Features such as commit check, rollback etc are added for easy manageability.
- Fine Grained Control: Thru the use of simple routing policies, administrators can have tighter control of the network traffic as well as gain advantages of having separate control and forwarding planes in place.

2.10 Junos Real-Time Performance Monitoring (Junos RPM)

The Junos RPM feature allows network administrators to measure performance parameters between two network endpoints. The RPM is a service-level monitoring tool. To collect network statistics RPM makes use of "RPM Probes." A network administrator can configure one endpoint to send targeted probes to either a destination IP address or URL. Once a response is received from the peer endpoint, statistics such as Round-Trip Time (maximum, minimum and average), standard deviation, jitter, probe counts can be ascertained. Another advantage of Junos RPM is its ability to set Quality of Service markings on test probes. Junos RPM can also be used to track if BGP neighbors are active [42].

2.11 Virtual SRX (vSRX) (Formerly called Firefly Perimter)

To capitalize on the benefits of virtualizations, many enterprises are moving their workloads to the cloud [36]. Juniper Networks, a key player in the networking industry offers a virtual next-generation firewall solution called vSRX (formerly Firefly Perimeter). The commercial version of this product offers speeds of up to 100 Gbps. vSRX can also perform stateful firewall protection, VPN functionality, screen-options, zone and address book creation and other network capabilities. Thru a simple command, this virtual firefly can be converted into a virtual router offering complete routing capabilities. Some of the protocols vSRX can run are OSPF, IS-IS, RIP, BGP, MP-BGP, MPLS, RSVP, LDP as well as a number of other protocols. vSRX supports KVM and VMware hypervisors [43] [44] [45].

2.12 Colasoft Packet Player

The Colasoft packet player is program which allows individuals to open packet captures and packet trace files and replay the captured packets back into the network. Wireshark, Colasoft Capsa, Etherpeek, network general sniffer etc. packet captures are supported by this program. Another advantage of this program is it lets users select the play speed as well loop packet captures while creating a delay between successive loops. The burst mode lets users send packets instantly without any delay. The program is offered in a free and paid version [46].

2.13 Tcpeplay and Tcprewrite

Tcpreplay is a suite of tools licensed under GPLv3 written by A. Turner for the UNIX operating system. It allows users to replay captured packets back onto the network through routers, switches, firewalls, IDS and IPS devices. One of the tools within this broad suite is

tcprewrite. This tool allows users to rewrite layer 2, layer 3, layer 4 and certain layer 5 to 7 packet information [47] [48]

2.14 Wireshark

A project which was initially started in 1998 by Gerald is today the de facto standard for packet capture and packet analysis tool for government, educational and non-profit enterprises. Wireshark allows deep packet analysis of over a hundred protocols in real time as well as offline modes. The application also boosts of a number of display filters helping analyst look for specific details in traffic flows. The software also allows decryption support for popular protocols such as IPSec, ISAKMP, SNMPv4, SSL/TLS and WPA2. After completing packet captures, wireshark allows users to save their output files in many popular formats such as Pcap NG, tcpdump, Etherpeek etc [49].

2.15 FileZilla

FileZilla is a free FTP solution which is available both as a client application and server application. This application is an open source software. The software supports FTP, SFTP and FTPS. The simple GUI lets users drag and drop files between the client and server. FileZilla is available for Windows, Linux and Mac OS X and also supports IPv6 [50] [51].

2.16 Microsoft Visio

Formerly called Microsoft Office Visio, Microsoft Visio is an application from Microsoft Corporation which enables users to create advanced network diagrams, flowcharts, organization charts and engineering designs using modern shapes and templates. The software also allows increased visibility for enterprise customers by allowing them to share their process models and collaborate thru a browser [52].

2.17 Microsoft Excel 2016

Microsoft Excel 2016 is an application with the Microsoft Office Suite from Microsoft Systems. This software allows users to input data on spreadsheets, organize, format and calculate data with the help of formulas. The software also performs complex analysis for the user my summarizing data into pivot-tables. Additionally, Microsoft Excel also enables users to visualize their data in graphs and pie-charts [53].

2.18 iTerm2

iTerm2 is licensed under GPL v2. The software is the successor to iTerm. This software is typically used as a replacement for the Terminal application found on Mac OSX. The software offers useful features such as split plane view allowing a multi-session work flow. Another interesting feature is the ability to search and find text using regular expressions. iTerm 2 also lets users create logging sessions to log the contents displayed on screen to an external log file [54].

2.19 TextEdit

TextEdit is a text editing application on Mac OSX developed by Apple Inc. This application comes preloaded on Mac OSX systems. The application is based on the text system in Cocoa. Being an open source word processor, the application has several useful tools. The application by default saves documents in Rich Text Format but also supports several other popular file extensions such as Word (1997 – 2007), html and odt. [55].

2.20 OS X El Capitan

OS X El Capitan is an operating system from Apple Inc. which runs on desktops and servers. OS X El Capitan is the twelfth release of OS X. The name of this operating system

was derived from a famous rock formation in Yosemite National Park, USA. Several improvements have been made from its predecessor OS X Yosemite. The main areas where improvement was made are performance, security, design and usability [56].

2.21 Ubuntu 14.04.02

Ubuntu is a Debian-based Linux operating system. This operating system has been developed by Canonical Ltd. Ubuntu is also based on open-source software. The operating system can be run on a varied number of devices from low memory IoT devices to high performance networks such as the Cloud. Security is also very important to the Ubuntu OS and the operating system offers a built-in firewall. Ubuntu's operating system also does not require any licenses and offers regular updates free of charge [57].

2.22 Microsoft Windows XP (Service Pack 3)

Windows XP is a personal computing operating system from Microsoft. The initial release of Microsoft Windows XP was in 2001. Service Pack 3 was released in 2008. The update fixed over 1174 bugs and include several other important new features such as support for SHA-2 signatures in X.509 certificates, Network Access Point client etc. In 2014, Microsoft ended the extended support for this operating system [58].

3 Network Model and Implementation

A number of simulators both commercial and open source are available to conduct the tests required to prove the initial hypothesis and performance of MPLS in a virtualized network. Some of these simulators include but are not limited to OPNET, Cloonix, Core, GNS3, OFNet, UNetLab, Cisco VIRL [59].

However, since this thesis required carrier-grade routers I decided to use a commercial carrier grade router from Juniper Networks called Firefly Perimeter running on VMware Fusion as the chosen hypervisor. Additional details about the test bed and implementation can be found below.

3.1 Building the Test Bed

The following subsections explain how the test bed is built.

3.1.1 Preparing the Host System, Hypervisor & Virtualization Guest OS

VMware Fusion (version 7.1.3) has been chosen as the hypervisor for this thesis. The same has been downloaded via a student account from Rochester Institute of Technologies – Golisano College of Computing and Information Sciences – Information Sciences and Technology portal [60].

Figure 3.1: VMware Fusion Version

VMware Fusion was installed on a MacBook Pro, running the following specifications:

- Processor – 2.3GHz Intel Core i7
- Memory – 16 GB 1600 MHz DDR3
- Operating System – OS X El Capitan Version 10.11.6

Hardware Overview:

Model Name:	MacBook Pro
Model Identifier:	MacBookPro11,3
Processor Name:	Intel Core i7
Processor Speed:	2.3 GHz
Number of Processors:	1
Total Number of Cores:	4
L2 Cache (per Core):	256 KB
L3 Cache:	6 MB
Memory:	16 GB
Boot ROM Version:	MBP112.0138.B17
SMC Version (system):	2.19f12

Figure 3.2: MacBook Pro Specifications

Next, an evaluation image of vSRX formerly called Juniper Networks Junos Firefly Perimeter (junos-vsrx-12.1X47-D15.4-domestic.ova) was download from the Juniper website [61]. Although this image is popularly used as a firewall, it can also be used as a virtual router running MPLS services. Juniper Networks carrier grade router vMX could also be

used, however this would impose scalability issues due to the hardware limitations. Further information on the same, can be found in the Future Direction section of this thesis.

Given below are the settings configured for the thirteen virtual routers.

- Processor: 2 Processor Cores
- Memory: 1200 MB
- Accelerate 3D Graphics: Disabled

Kindly note, these are not the recommended settings from Juniper Networks. However, for the purpose and scalability of this thesis model the settings configured are adequate.

Apart from the above images, an evaluation copy of Microsoft Windows XP SP3 was used as the source of Traffic Generation for Voice, Video and Data traffic. The reason Windows XP was used is due to the low system requirements needed by this operating system. This also allowed me to scale my network further by allocating additional resources to the nodes within the Service Provider Core. The system settings given to this virtual machine are given below:

- Processor: 1 Processor Core
- Memory 280 MB
- Number of Network Adapters: 3
- Applications Installed: Colasoft Packet Player (v1.3), Wireshark (v1.12.6)

To emulate the destination, the guest operating system used is Ubuntu 14.04.02 with the following specifications:

- Processor: 1 Processor Core
- Memory 280 MB
- Network Adapters: 1
- Applications Installed: Tcpreplay-tcprewrite [3.4.1]

3.1.2 Physical and Logical Topology

Keeping in mind the hardware limitations and after referencing several service provider designs, a small sized Service Provider Core Network was built to prove my hypothesis. The below Microsoft Visio Logical Topology, Figure 3.3 shows a Service Provider Core providing BGP L3 VPN Connectivity by connecting Customer-A's Site 1 and Site 2.

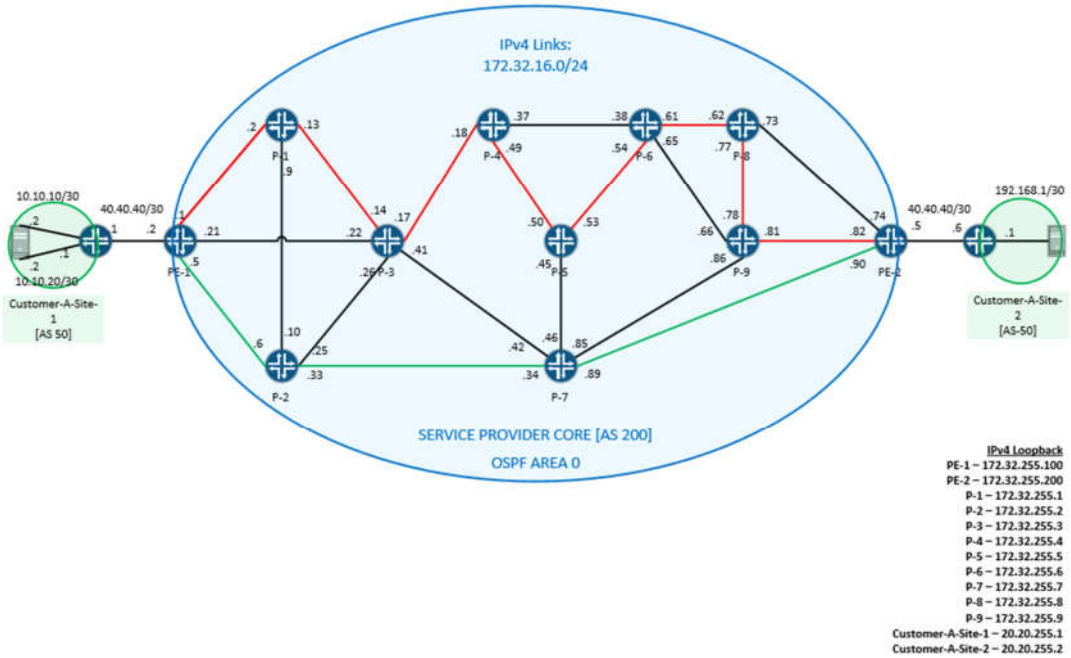

Figure 3.3: Logical Topology

Physical Connectivity between Virtual Routers and vmnet mappings can be seen in Figure 3.4 below:

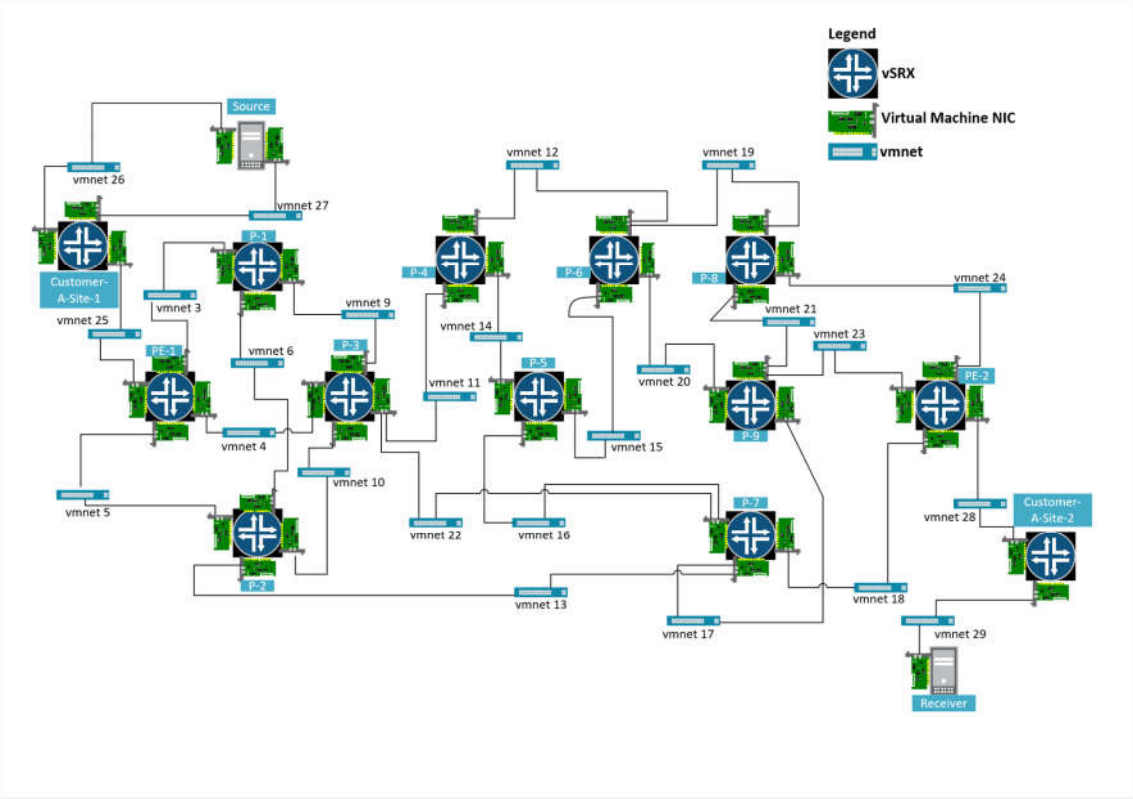

Figure 3.4: vmnet Mapping in VMWare Fusion

In order to connect the virtual routers VMware offers us VMNet functionality. Thirty-four vmnets were created and subsequently the virtual routers were connected by placing an interface from virtual router 1 to virtual router 2 on the same vmnet. This essentially creates a logical connectivity between the Guest OS's. Following which IPv4 address were assigned within the same network on both virtual routers to create layer 3 reachability.

3.1.3 Traffic Generators

To emulate real world traffic as close as possible, I used the following methods to generate voice, video and data traffic.

3.1.3.1 Voice Traffic

In order to emulate real-time voice traffic, a sample SIP call with RTP in G.711 was downloaded from Wireshark [62]. Next this packet capture was analyzed and certain changes were made in order to forward voice traffic thru the Service Provider Core. More information on these changes can be found in the section 3.1.4.1 of this thesis. After making the changes, the "Voice_Traffic.pcap" file was copied to the source server present in Customer-A-Site-1. Next, Colasoft Packet Player was opened and the above file was loaded choosing the correct Network Adapter (Network Adapter 1) as the egress interface on the virtual machine. A single instance of Colasoft Packet Player at burst play speed, ignoring file errors and with loops enabled would generate approximately 6.62 Mbps on the incoming interface of Customer-A-Site-1's ge-0/0/2 interface. Hence, I decided to run two instances simultaneously to generate enough traffic for all Scenarios. An example of the Packet Capture and Colasoft Settings can be seen in Figure 3.5 and Figure 3.6:

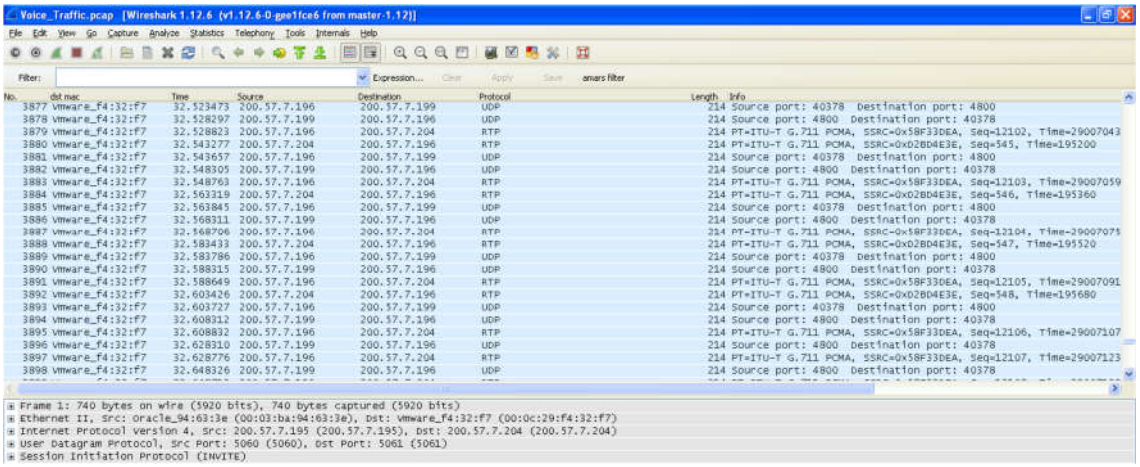

Figure 3.5: Wireshark Capture of Voice Traffic

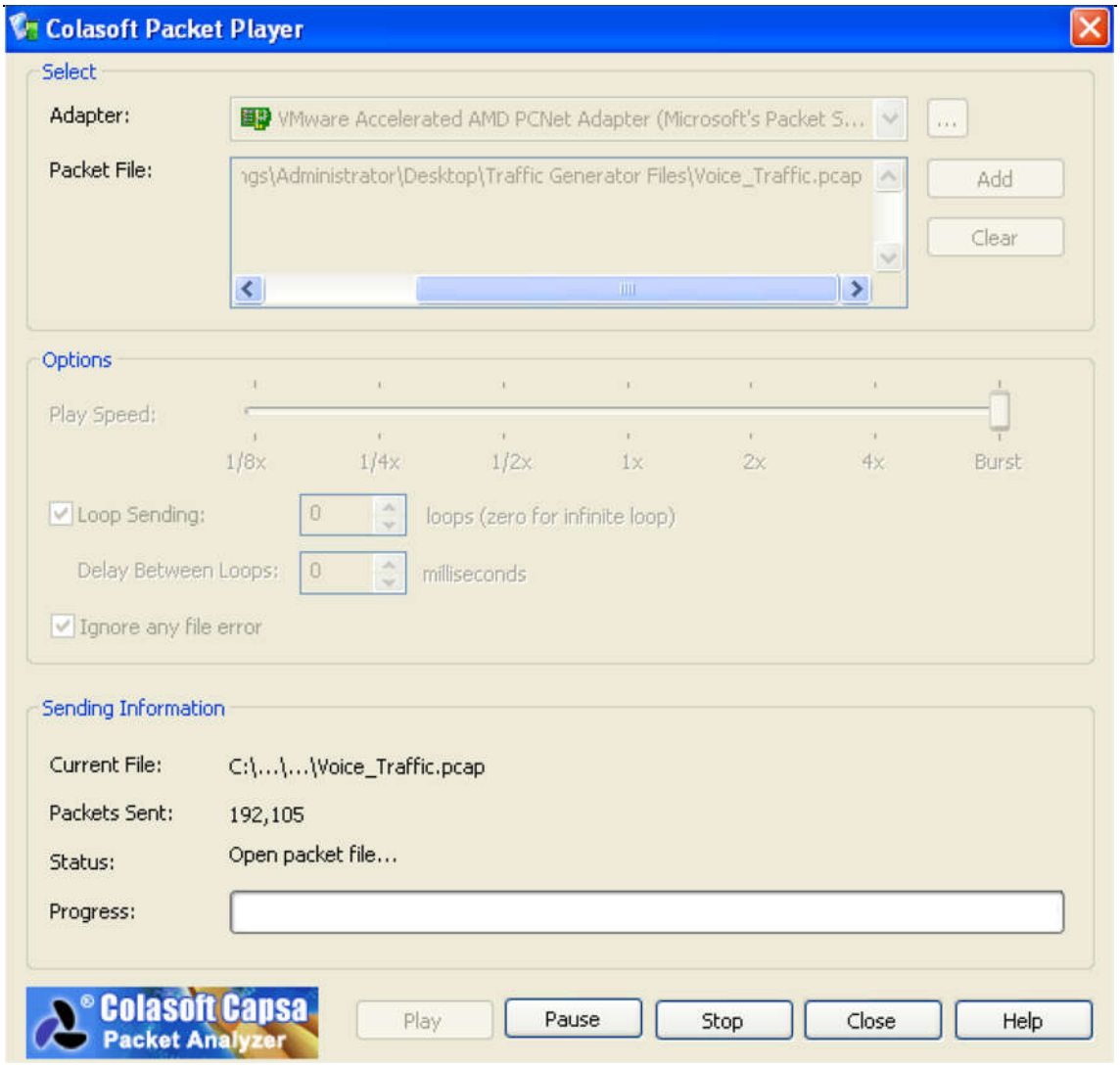

Figure 3.6: Colasoft Packet Player settings for Voice Traffic

3.1.3.2 Video Traffic

To simulate video traffic, VLC media player was used. A video file was added and then streamed from Site 1 to Site 2. At the same time a Wireshark instance to capture packets was running, capturing video traffic on the network adapter. The capture was saved as "Video_Traffic_Utkarsh". Next, this captured file was opened in Colasoft Packet Player. A single instance of video traffic at burst play speed, ignoring packet errors enabled would generate approximately 65.27 Mbps on the incoming interface of Customer-A-Site-1's ge-

0/0/3 interface. An example of the Packet Capture and Colasoft Settings can be seen in Figure 3.7 and Figure 3.8:

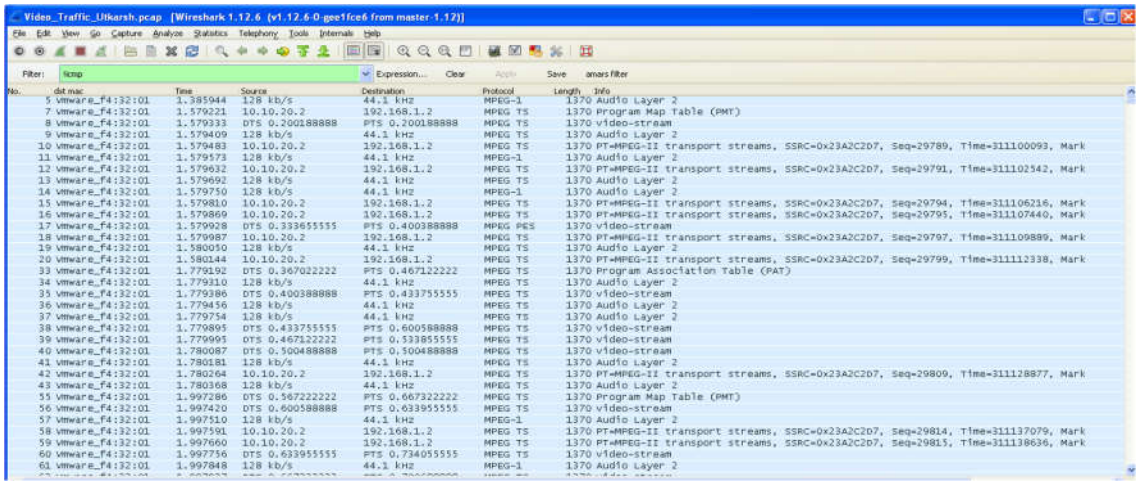

Figure 3.7: Wireshark Capture of Video Traffic

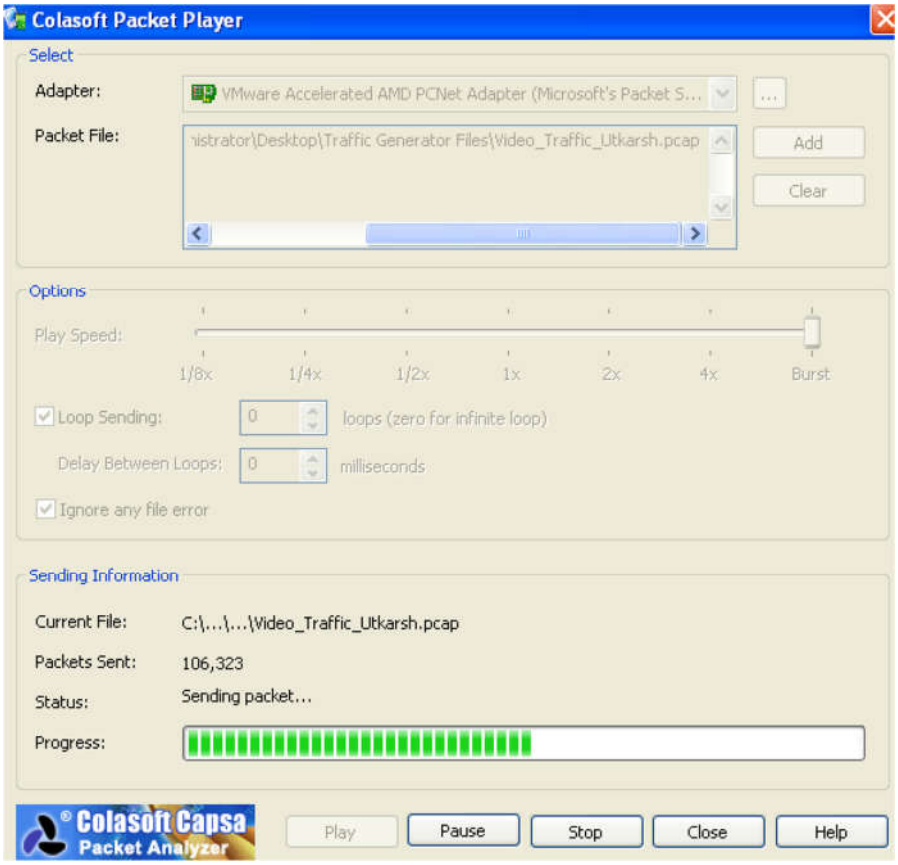

Figure 3.8 Colasoft Packet Player settings for Video Traffic

3.1.3.3 Data Traffic

To simulate data traffic, a Wireshark capture was running on Network Adapter 3. A dummy file was created on Customer-A-Site-2 with the help of logging in to the shell mode of CentOS router. The command given was, “dd if=/dev/zero of=download.file bs=204800000 count=1”. Next FTP was enabled on the router and the above file was pulled and saved via FTP FileZilla Software installed on the Source Server.

Next, the Wireshark capture was started, capturing all packets flowing on Network Adapter 3, the adapter which will be used to send receive FTP Data. An upload was made from Customer-A-Site-1’s source server onto Customer-A-Site-2’s virtual router. This upload was captured and the resultant file is saved as “Data_Traffic.pcap”.

Finally, Colasoft Packet Player was opened and a single instance of data traffic at burst play speed, ignoring packet errors enabled would generate approximately 60.8 Mbps on the incoming interface of Customer-A-Site-1’s ge-0/0/3 interface. An example of the Packet Capture and Colasoft Settings can be seen in Figure 3.9 and Figure 3.10:

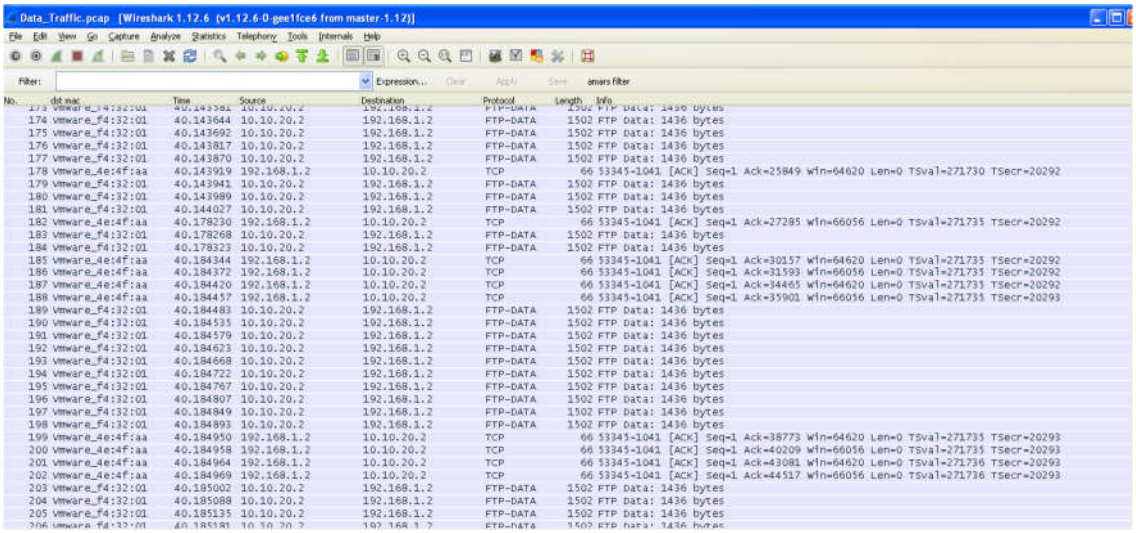

Figure 3.9: Wireshark Capture of Data Traffic

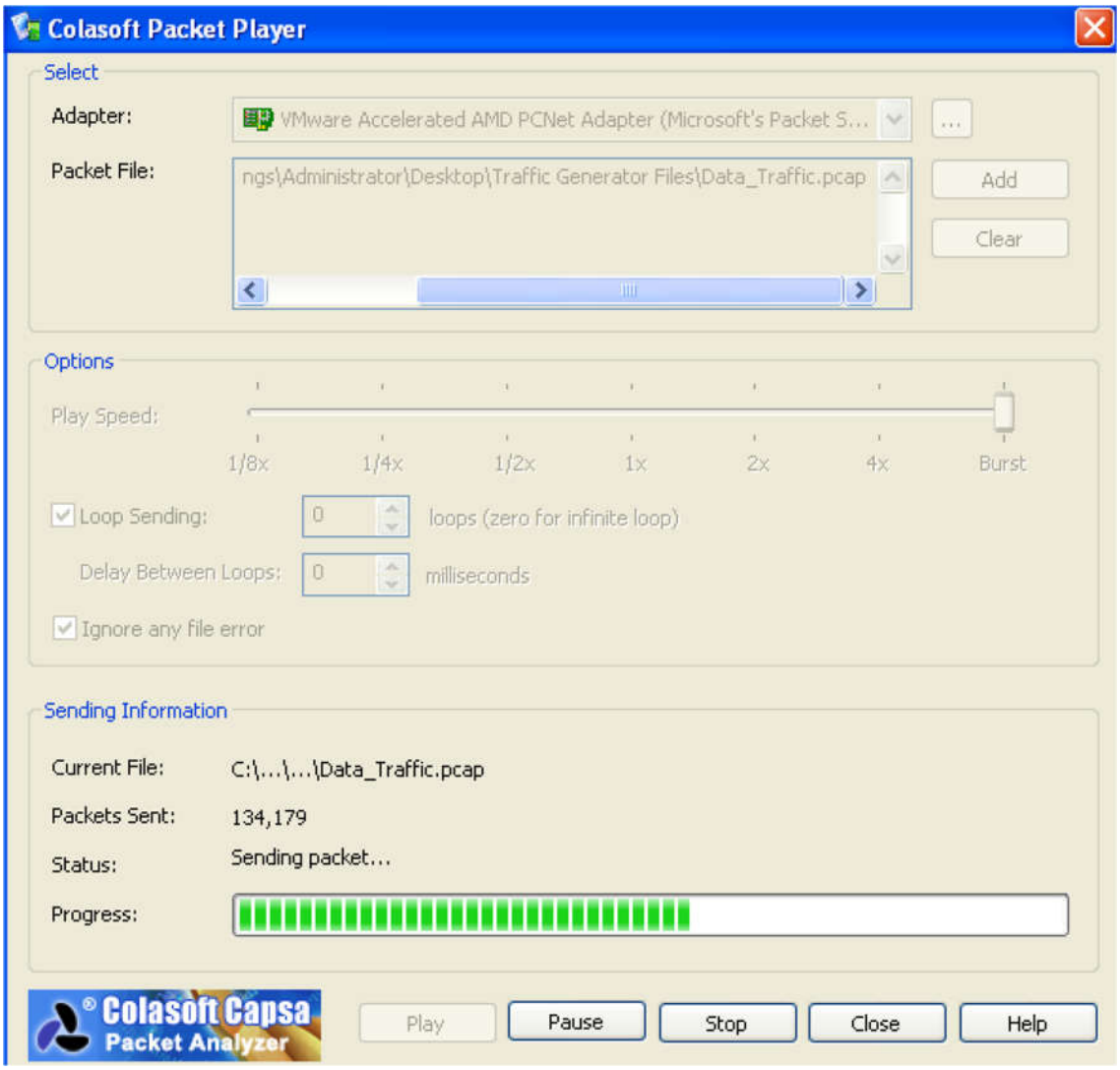

Figure 3.10 Colasoft Packet Player settings for Data Traffic

3.1.4 Provisioning Customer-A Site 1 and Site 2

3.1.4.1 Customer-A Site 1

To provision site 1's source server, an evaluation copy of Microsoft Windows XP SP3 was used. The main reason for using this OS was its light system requirements. Next, a challenge was faced to separate voice, video and data traffic. The voice traffic was generated

using a pre-captured file and since the destination IP address within the packets were different, I needed a way to classify them as voice based on some similarity.

To classify the traffic as voice, voice traffic got its own egress network adapter connecting to the ge-0/0/2 interface of Customer-A-Site-1's Virtual Router. Hence, I decided to rewrite the destination mac address of all packets in the voice packet capture to have the destination mac address of the ge-0/0/2 interface of Customer-A-Site-1's virtual router using a utility called tcprewrite present in Tcpreplay package installed on the Ubuntu Guest OS. The command given to change the destination mac address is "tcpwrite --infile=SIP_CALL_RTP_G711.pcap --output=Voice_Traffic.pcap --enet-dmac=00:0c:29:f4:32:f7" This command rewrites the destination mac address of all packets from the "SIP_CALL_RTP_G711.pcap" and saves the new packets to "Voice_Traffic.pcap". The new destination mac address given is shown in the above command, which is "00:0c:29:f4:32:f7". This mac address was statically configured on Customer-A-Site-1's incoming voice interface, ge-0/0/2. A firewall filter has been applied to this interface, which classifies incoming traffic into the Voice Forwarding Class (Voice_EF). The vSRX has been configured to do egress QoS. The rewrite rule configured for all traffic in the "Voice_EF" is using inet-precedence markings set. The Most Significant Bits (MSB) in the Type of Service field have been rewritten to "101". The same can been seen in the configurations files present in the Appendix chapter of this thesis. Thereafter, a default route is created on Customer-Site-2 and pushed via BGP to the Service Provider Core, which in turn forwards it to Customer-A-Site-1's customer edge router. The reason for doing this is to carry voice traffic, with destinations address that do not exist in our topology thru the service provider core out onto the remote site.

Video and Data traffic are pushed from the source VM, i.e. via Colasoft Packet Player using Network Adapter 3. This adapter has the default gateway set as Customer-A-Site-1's

virtual routers ge-0/0/3 interface. To separate data and video traffic, a stateless firewall filter has been used, which classifies all traffic having a destination port of 5004 being classified as video traffic with all remainder traffic as data traffic. Once the traffic was classified with the help of the firewall filter, video traffic was placed into (Video_AF) forwarding class, while data traffic was placed into the "DATA_BE" forwarding class. When traffic exits Customer-A-Site-1's virtual router, traffic is being rewritten with inet-precedence having the following ToS bits set:

- Data Traffic – 000
- Voice – 101
- Video - 010

Finally, in order to find the Round-Trip Time Junos RPM probes are being sent from Site 1 to Site 2. These probes are kept in the correct forwarding classes with the help of a stateless firewall filter configured on the "ge-0/0/1" incoming PE-1 virtual router. As these probes are sent from Site-1 to Site-2, they are given the following DSCP markings:

- Data RPM Probes: 000000
- Voice RPM Probes: 101110
- Video RPM Probes: 001010

3.1.4.2 Customer-A Site 2

Customer-A-Site 2 is the receiver of voice, video and data traffic. An Ubuntu Guest OS is used as the receiver's server. Thereafter, a default route is created on Customer-Site-2's Customer Edge router and pushed via BGP to the Service Provider Core, which in turn forwards it the Customer-A-Site-1's site. The benefit of doing this is to push voice traffic from site-1, for destinations address that do not exist in our topology thru the service provider core out onto Site 2.

3.1.5 Provisioning the Service Provider Core

Within the Service Provider Core, 11 instances of vSRX were configured and started. The configuration can be found in the appendix section of this thesis. Two vSRX's were given the role of Provider Edge devices (PE-1, PE-2) which were in charge of housing the customers virtual routing and forwarding tables, while at the same time being the Ingress and Egress nodes for the MPLS label switched path.

Once the underlying physical topology was completed using the "vmnet" mappings mentioned in section 3.1.2, IP address were assigned on the gigabit interfaces. Interface speeds were also reduced to 100 Mbps and were configured to operate in full-duplex mode. The IP addresses assigned can be seen in figure 3.3. The underlying interior gateway protocol used within the Service Provider Core is Open Shortest Path First version 2 (OSPFv2). All service provider virtual routers are placed in a single area OSPF implementation, area 0. Following which all multi-access interfaces were configured as point-to-point interfaces under the OSPF configuration to save time during adjacency formation as point-to-point links negate the need of electing routers as a "designated router" or "backup designated router". To aid in faster detection of link failure and alert the IGP to use an alternate path, "Bidirectional Forwarding Detection (BFD)" is used. BFD hellos are sent across all service provider links configured for OSPF operations. BFD hellos are being sent and received every 1.2 seconds. The default Junos multiplier of 3 is being used, wherein if the virtual router fails to receive a single BFD hello within 3.6 seconds (1.2 seconds x 3 BFD Rx hellos), BFD alerts OSPF not to wait for its neighbor's hello packet and flag the adjacency as Down.

The remaining nine instances were given the role of Provider devices, providing transit services for the MPLS traffic. Further information can be found below:

3.1.5.1 Provider-Edge 1 (PE-1)

PE-1 is directly connected with Customer-A-Site-1 and is using an eBGP session to exchange routes. In order to store the customer routes from the local and remote site, a special routing instance called “virtual routing and forwarding (VRF)” instance has been created on the local PE. Within this instance, the “as-override” feature is also used. This allows the local PE to send routes received from the remote site to the local site. The VRF enables complete isolation for the customers IPv4 unicast routes by storing them in a unique routing table called “Customer-A.inet.0”.

Since the PE would also serve as an ingress router for traffic being pushed to the remote site and as an egress router for traffic being sent to the local customer site, “Multiprotocol Label Switching (MPLS)” protocol has been enabled on all interfaces facing the Service Provider core. Next, “Resource Reservation Protocol (RSVP)” was configured and with the help of “Explicit Route Objects (EROs)” path messages were signaled to build the primary and secondary paths for the label switched path (LSP) called “PE-1-to-PE-2”. The primary MPLS LSP path is called “Best_Path” and has been signaled from PE-1 being the ingress router, while P-2 and P-7 acting as transit label switched routers while finally egressing on PE-2. On the other hand, the secondary LSP path starts from PE-1 being the ingress node and PE-2 being the egress while transiting thru P-1, P-3, P-4, P-5, P-6, P-8 and P-9 respectively. The secondary path is configured as “Failover_Path”. By default, the MPLS LSPs would use the entire links bandwidth (i.e. 100 Mbps) and therefore a configuration was made to police traffic in excess of 10 Mb. Figure 3.11 shows the Primary and Secondary paths. The primary path transverses the green links whereas the red links symbolizes the secondary path.

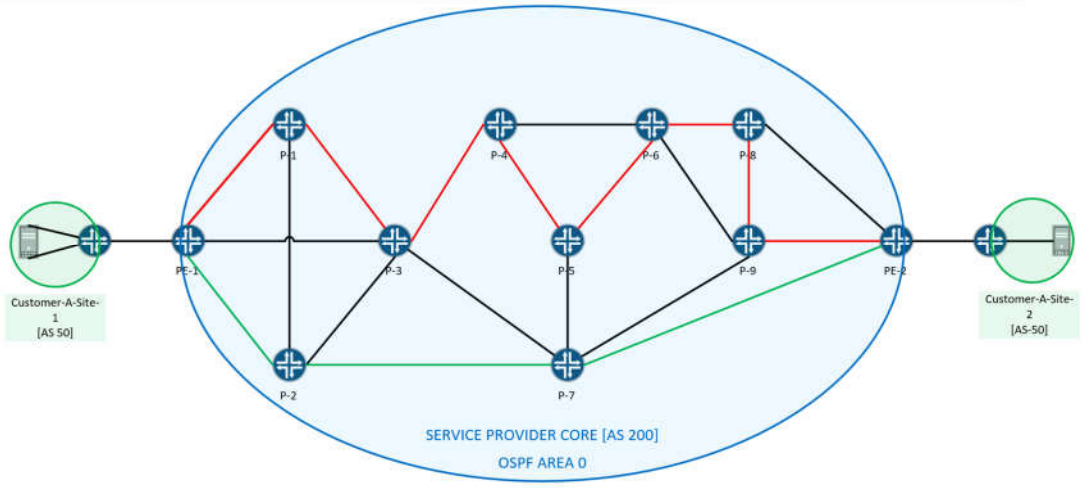

Figure 3.11: MPLS LSP Primary and Secondary Path

A Border Gateway Protocol (BGP) session has been signaled from PE-1 to PE-2 which is configured with "family-inet" for Internal Border Gateway Protocol (iBGP) reachability. To carry the Route Distinguisher and Route Target, "family-inet-vpn" family has been signaled, which enables Multi-Protocol Border Gateway Protocol (MP-BGP) functionality. Appropriate VRF import and export policies are also in place to accept and send routes with the correct communities.

On interface, "ge-0/0/1", which is the incoming interface for all traffic received from the local customer site a stateless firewall filter was configured. This firewall filter is used to allocate the Junos RPM probes and Customer-A-Site-1's voice, video and data traffic into their respective forwarding classes and exit the router with the correct EXP rewrite rules. Below is the breakout of the markings used by PE-1 to decide how traffic should be treated.

- RPM Probe – QoS Classifiers
    - Voice – DSCP EF
    - Video - DSCP AF11
    - Data – DSCP BE
- Traffic Classifiers

- Voice – Inet-Precedence Critical-ECP Bits
- Video – Multifield classifier matching all traffic with destination-port 5004
- Data – Any remaining traffic not matching any of the above classifications is treated as data traffic.

In certain scenarios (explained in detail within Section 3.4), Fast Reroute has been enabled. In order to optimize the Fast-Reroute feature, an export policy is configured which places both the primary and backup next-hops in the forwarding table.

3.1.5.2 Provider-Edge-2 (PE-2)

PE-2 is directly connected with Customer-A-Site-2 and is using an eBGP session to exchange routes. In order to store the customer routes from the local and remote site, a special routing instance called "virtual routing and forwarding (VRF)" instance has been created on the local PE. Within this instance, the "as-override" feature is also used, allowing the local PE to send routes received from the remote site to the local site. The VRF enables complete isolation of the customers IPv4 unicast routes by storing them in a unique routing table called "Customer-A.inet.0". Since the PE would also serve as an ingress router for traffic being pushed to the remote site and as an egress router for traffic being sent to the local customer site, "Multiprotocol Label Switching (MPLS)" protocol has been enabled on all interfaces facing the Service Provider core. Next, "Resource Reservation Protocol (RSVP)" was configured and with the help of "Explicit Route Objects (ERO's)" path messages were signaled to build the primary and secondary paths for the label switched path (LSP) called "PE-2-to-PE-1". The primary MPLS LSP path is called "Best_Path" and has been signaled from PE-2 being the ingress router, while P-7 and P-2 acting as transit label switched routers while finally egressing on PE-1. On the other hand, the secondary LSP path

starts from PE-2 being the ingress node and PE-1 being the egress while transiting thru P-9, P-8, P-6, P-5, P-4, P-3 and P-1 respectively and is called “Failover_Path”. By default, the MPLS LSP would use the entire links bandwidth (i.e. 100 Mbps) and therefore a configuration was made to police traffic in excess of 10 Mb.

A Border Gateway Protocol session has been signaled from PE-2 to PE-1 which is configured with “family-inet” for Internal Border Gateway Protocol (iBGP) reachability and additionally to carry the Route Distinguisher and Route Target, “family-inet-vpn” family has been signaled, which enables Multi-Protocol Border Gateway Protocol functionality. Appropriate VRF import and export policies are also in place to accept and send routes with the correct communities.

In certain scenarios (explained in detail with Section 3.4), Fast Reroute has been enabled. In order to optimize the Fast-Reroute feature, an export policy is configured which places both the primary and backup next-hops in the forwarding table.

3.1.5.3 Provider Routers (P-1 to P-9)

All the Provider routers, i.e P-1 to P-9 have similar configuration in place. All these routers are configured to run OSPF in Area 0. MPLS and RSVP have also been enabled to build the MPLS transport layer. Some additional configuration is made on P-2 and P-7 which places all its links in a protected state. This enables OSPF to calculate a loop-free-alternate path (LFA), which is used by the Fast Re-Route Protection feature. Configuration of the node-link-feature is only done on these two nodes as the link between these nodes would be made “down” in order to failover traffic from the primary path to secondary path.

3.2 Testing Tool – Junos Resource Performance Management (Junos RPM)

The performance metric chosen for this thesis is Round Trip Time. Many tools are available to ascertain the above metric. However, an inbuilt functionality in Junos is present

called Junos Resource Performance Management (Junos RPM) which can provide Round-Trip Time values. The reason this method was chosen, is due to a low processing overhead on the virtual routers. Also, Junos RPM can be initiated directly from Customer-A-Site-1's router to Customer-A-Site-2's router. Another, major advantage of Junos RPM is its ability to mark the RPM Probes with QoS settings. Using the below marking's RPM probes are generated from Customer-A-Site-1's virtual router to Customer-A-Site-2's loopback address. The markings can be found below:

- Voice RPM Probes – DSCP EF
- Video RPM Probes - DSCP AF11
- Data RPM Probes – DSCP BE

A probe is sent every second from Site-1 to Site-2. Probes are sent with the following data sizes, to match the average payload size of the user traffic:

- Voice – 214 Bytes
- Video - 1370 Bytes
- Data – 1436 Bytes

Though all tests require the use of only three hundred probes, a history buffer of six hundred probes has been configured. As mentioned earlier, the RPM probes will be placed in the correct queues by PE-1 using a stateless firewall filter. Additional information regarding the commands and analysis of the probes can be found in Section 3.4 of this thesis.

3.3 Simulation Scenarios

VMware Fusion v7.1.3 is the chosen hypervisor to run the below simulations. All virtual routers are running Junos OS version "junos-vsrx-12.1X47-D15.4-domestic". Junos RPM Probes for all the below scenarios are being sent from Customer-A-Site-1's virtual-

router to Customer-A-Site-2's virtual-router. To prove the initial hypothesis and answer the research questions the below 8 scenarios have been tested and the collected results have been gathered and analyzed. Figure 3.12, shows us the topology being used to run the simulations. As discussed above, Customer-A-Site-1 has the source traffic generator whereas Customer-A-Site-2 is the traffic receiver. A Label Switched Path is created from PE-1 to PE-2 and for return traffic another Label Switched Path has been made from PE-2 to PE-1. Each label switched path has two paths. The primary path is called "Best_Path" and is denoted via the green line in Figure 3.12. On the other hand, the secondary path is called "Failover_Path" and is denoted via the red line.

For each test in the below scenarios, RPM probes would be sent on a per second basis from Customer-A-Site-1 to Customer-A-Site-2. Depending on the scenarios, the Junos RPM probes would enter specific queues (QoS Enabled Sceanrios) or be categorized as "Best_Effort" (QoS Disabled scenarios) with the help of a firewall filter configured on PE-1's ge-0/0/1 interface called "rpm-classifier". These probes would additionally enable us to track the Round-Trip Time for traffic between the two sites while entering the virtualized Service Provider Core. Traffic would initially be sent on the "Best_Path", which tends to be the least hop count / least cost path in the IGP domain. In a real-world scenario, the primary label-switched-path would be preferred over any alternate paths available. After approximately, 150 probes (150 seconds), the link between P-2 and P-7 would be brought down, thus initiating a failover from the "Best_Path" to the "Failover_Path". Doing so allows us to understand how Fast Reroute performs in a virtualized Service Provider Core.

Apart from testing the performance of Fast Reroute, I have also compared scenarios for voice, video and data traffic being sent at 8 Mb and 12 Mb on a 10 Mb LSP. Doing so, enables me to get a better understanding of QoS in a virtualized Service Provider core network.

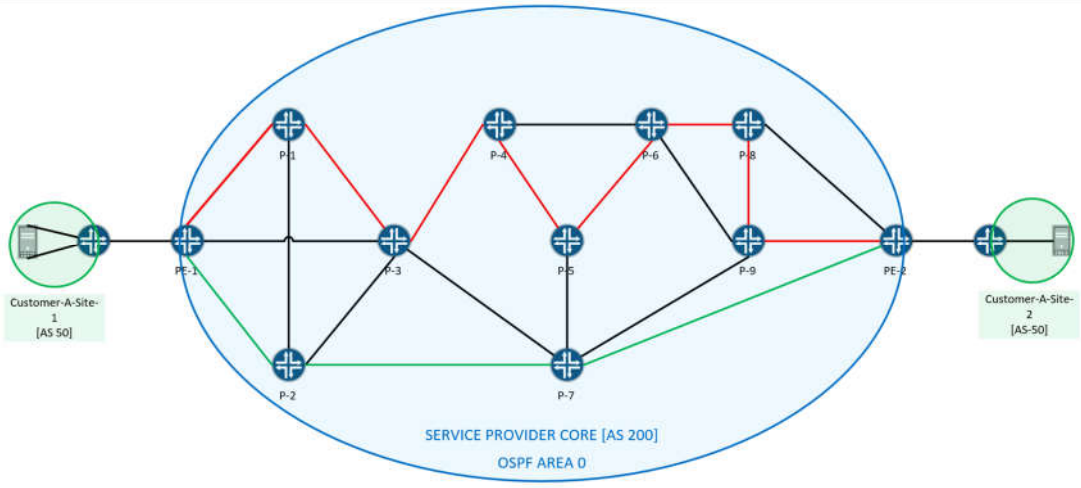

Figure 3.12: MPLS LSP Primary and Secondary Path

Four firewall filters are configured on Site1's CE virtual router. These firewall filters in turn call policers which rate limit traffic being sent from the source. The configuration of these filters can be found in the appendix section of this thesis. However, for the purpose of understanding how traffic is sent at either 8 Mbps or 12 Mbps from the Site-1's CE virtual router to PE-1, I have explained briefly what each firewall filter does. The firewall filters name and a brief is description is given below:

- Filter "normal-dav" – This firewall filter is used to rate limit video and data traffic. Video traffic is separated from data traffic by matching traffic having a destination-port number of "5004" (All video packets which were captured from VLC Media player have this source number set) and checked against a policer (normal-video) which allows 4 Mb of Video traffic and an additional burst of 15000 bytes. Video traffic exceeding the bandwidth would be dropped. Any traffic which does not have a destination port of "5004" is classified as data traffic and checked against another policer (normal-data) which permits 2 Mbps and an additional burst size of 15000 bytes. Additional data traffic is dropped.

  description is given below:

- Filter "normal-voice" – This firewall filter is used to rate limit voice traffic. All traffic sent to this policer (normal-voice), permits 2 Mbps and an additional burst size of 15000 bytes. Additional voice traffic is dropped.
- Filter "congested-dav" – This firewall filter is used to rate limit video and data traffic. Video traffic is separated from data traffic by matching traffic having a destination-port number of "5004" (All video packets which were captured from VLC Media player have this source number set) and checking it against a policer (congested-video) which allows 6 Mb of Video traffic and an additional burst of 15000 bytes. Video traffic exceeding the bandwidth would be dropped. Any traffic which does not have a destination port of "5004" is classified as data traffic and checked against another policer (congested-data) which permits 3 Mbps and an additional burst size of 15000 bytes. Additional data traffic is dropped.
- Filter "congested-voice" – This firewall filter is used to rate limit voice traffic. All traffic is sent to a policer (congested-voice), which permits 3 Mbps and an additional burst size of 15000 bytes. Additional voice traffic is dropped.

The firewall filters are activated based on the required bandwidth on a per scenario basis. Traffic from the source server is hitting Site 1's CE device at the following speeds using Colasoft Packet Player:

- Voice – 13.24 Mbps
- Video – 65.27 Mbps
- Data – 60.8 Mbps

The following lines explain the differences between all the eight scenarios while at the same time specify which sections of the configuration on the virtual routers have been modified:

3.3.1 Scenario 1: QoS and FRR Disabled – 8 Mb of Traffic on a 10 Mb LSP

This scenario is useful to understand how a virtualized Service Provider Core performs when the label switched path is uncongested, i.e. carrying traffic at a lower bandwidth than its maximum bandwidth carrying capacity with Quality of Service being disabled. The link tends to have an additional 2 Mb of unused bandwidth. Round Trip Time can be used as a suitable metric, to illustrate the end to end latency. In addition to the above reason, I also wanted to understand how a virtualized service provider core would perform with Fast Reroute being disabled during failover from the primary path onto the secondary path. Junos RPM probe timeouts would help me understand the packet losses. The firewall filter enabled on Customer-A-Site-1's CE virtual router are "normal-dav" and "normal-voice". Below are the modifications made in the configuration for this scenario.

Configuration modifications required to disable Quality of Service

- Deactivate firewall filter "rpm-classifier" on PE-1's CE-PE link
- Deactivate Quality of Service configuration on PE-1
- Deactivate rewrite-rules on Customer-A-Site-1's PE-CE link, thus placing all traffic into the Best_Effort class.

Configuration modifications required to disable Fast Reroute

- Disable CSPF on all virtual-routers in the Service Provider core
- Activate path "Best_Path" and "Failover_Path" on PE-1 virtual router
- Deactivate Fast Reroute under MPLS configuration on PE-1 virtual router
- Deactivate Load Balancing Policy on PE-1
- Deactivate OSPF Traffic-Engineering extension on all virtual routers in the Service Provider core
- On P-2 and P-7 disable node-link-protection

3.3.2 Scenario 2: QoS and FRR Disabled – 12 Mb of Traffic on a 10 Mb LSP

This scenario is useful to understand how a virtualized Service Provider Core performs when the label switched path is congested, i.e. carrying traffic at its maximum bandwidth carrying capacity with Quality of Service being disabled. Round Trip Time can be used as a suitable metric, to illustrate the end to end latency. In addition to the above reason, I also wanted to understand how a virtualized service provider core would perform with Fast Reroute being disabled during failover from the primary path onto the secondary path. Junos RPM probe timeouts would help me understand the packet losses. The firewall filter enabled on Customer-A-Site-1's CE virtual router are "congested-dav" and "congested-voice". Below are the modifications made in the configuration for this scenario.

Configuration modifications required to disable Quality of Service

- Deactivate firewall filter "rpm-classifier" on PE-1's CE-PE link
- Deactivate Quality of Service configuration on PE-1
- Deactivate rewrite-rules on Customer-A-Site-1's PE-CE link, thus placing all traffic into the Best_Effort class.

Configuration modifications required to disable Fast Reroute

- Disable CSPF on all virtual-routers in Service Provider core
- Activate path "Best_Path" and "Failover_Path" on PE-1 virtual router
- Deactivate Fast Reroute under MPLS configuration on PE-1 virtual router
- Deactivate Load Balancing Policy on PE-1
- Deactivate OSPF Traffic-Engineering extension on all virtual routers in the Service Provider core
- On P-2 and P-7 disable node-link-protection

3.3.3 Scenario 3: QoS Enabled and FRR Disabled – 8Mb of Traffic on a 10 Mb LSP

This scenario is useful to understand if any benefits are achieved when a virtualized Service Provider Core carries traffic on uncongested paths, i.e. carrying traffic at a lower bandwidth than its maximum bandwidth carrying capacity with Quality of Service being enabled. The link tends to have an additional 2 Mb of unused bandwidth. Round Trip Time can be used as a suitable metric, to illustrate the end to end latency. In addition to the above reason, I also wanted to understand how a virtualized service provider core would perform with Fast Reroute being disabled during failover from the primary path onto the secondary path. Junos RPM probe timeouts would help me understand the packet losses. The firewall filter enabled on Customer-A-Site-1's CE virtual router are "normal-dav" and "normal-voice". Below are the modifications made in the configuration for this scenario.

Configuration modifications required to enable Quality of Service

- Activate firewall filter "rpm-classifier" on PE-1's CE-PE link
- Activate Quality of Service configuration on PE-1
- Activate rewrite-rules on Customer-A-Site-1's PE-CE link, thus placing all traffic into the Best_Effort class.

Configuration modifications required to disable Fast Reroute

- Disable CSPF on all virtual-routers in Service Provider core
- Activate path "Best_Path" and "Failover_Path" on PE-1 virtual router
- Deactivate Fast Reroute under MPLS configuration on PE-1 virtual router
- Deactivate Load Balancing Policy on PE-1
- Deactivate OSPF Traffic-Engineering extension on all virtual routers in the Service Provider core
- On P-2 and P-7 disable node-link-protection

3.3.4 Scenario 4: QoS Enabled and FRR Disabled – 12Mb of Traffic on a 10 Mb LSP

This scenario is useful to understand how a virtualized Service Provider Core performs when the label switched path is congested, i.e. carrying traffic at its maximum bandwidth carrying capacity when Quality of Service has been enabled. Round Trip Time can be used as a suitable metric, to illustrate the end to end latency. In addition to the above reason, I also wanted to understand how a virtualized service provider core would perform with Fast Reroute being disabled during failover from the primary path onto the secondary path. Junos RPM probe timeouts would help me understand the packet losses. The firewall filter enabled on Customer-A-Site-1's CE virtual router are "congested-dav" and "congested-voice". Below are the modifications made in the configuration for this scenario.

Configuration modifications required to enable Quality of Service

- Activate firewall filter "rpm-classifier" on PE-1's CE-PE link
- Activate Quality of Service configuration on PE-1
- Activate rewrite-rules on Customer-A-Site-1's PE-CE link, thus placing all traffic into the Best_Effort class.

Configuration modifications required to disable Fast Reroute

- Disable CSPF on all virtual-routers in Service Provider core
- Activate path "Best_Path" and "Failover_Path" on PE-1 virtual router
- Deactivate Fast Reroute under MPLS configuration on PE-1 virtual router
- Deactivate Load Balancing Policy on PE-1
- Deactivate OSPF Traffic-Engineering extension on all virtual routers in the Service Provider core
- On P-2 and P-7 disable node-link-protection

3.3.5 Scenario 5: QoS Disabled and FRR Enabled – 8Mb of Traffic on a 10 Mb LSP

This scenario is useful to understand how a virtualized Service Provider Core performs when the label switched path is uncongested, i.e. carrying traffic at a lower bandwidth than its maximum bandwidth carrying capacity with Quality of Service being disabled. The link tends to have an additional 2 Mb of unused bandwidth. Round Trip Time can be used as a suitable metric, to illustrate the end to end latency. In addition to the above reason, I also wanted to understand how a virtualized service provider core would perform with Fast Reroute being enabled during failover from the primary path onto the secondary path. Junos RPM probe timeouts would help me understand the packet losses. The firewall filter enabled on Customer-A-Site-1's CE virtual router are "normal-dav" and "normal-voice". Below are the modifications made in the configuration for this scenario.

Configuration modifications required to disable Quality of Service

- Deactivate firewall filter "rpm-classifier" on PE-1's CE-PE link
- Deactivate Quality of Service configuration on PE-1
- Deactivate rewrite-rules on Customer-A-Site-1's PE-CE link, thus placing all traffic into the Best_Effort class.

Configuration modifications required to enable Fast Reroute

- Enable CSPF on all virtual-routers in Service Provider core
- Activate path "Best_Path_cspf" and "Failover_Path_cspf" on PE-1 virtual router
- Activate Fast Reroute under MPLS configuration on PE-1 virtual router
- Activate Load Balancing Policy on PE-1
- Activate OSPF Traffic-Engineering extension on all virtual routers in the Service Provider core
- On P-2 and P-7 enable node-link-protection

3.3.6 Scenario 6: QoS Disabled and FRR Enabled – 12Mb of Traffic on a 10 Mb LSP

This scenario is useful to understand how a virtualized Service Provider Core performs when the label switched path is congested, i.e. carrying traffic at its maximum bandwidth carrying capacity with Quality of Service being disabled. Round Trip Time can be used as a suitable metric, to illustrate the end to end latency. In addition to the above reason, I also wanted to understand how a virtualized service provider core would perform with Fast Reroute being enabled during failover from the primary path onto the secondary path. Junos RPM probe timeouts would help me understand the packet losses. The firewall filter enabled on Customer-A-Site-1's CE virtual router are "congested-dav" and "congested-voice". Below are the modifications made in the configuration for this scenario.

Configuration modifications required to disable Quality of Service

- Deactivate firewall filter "rpm-classifier" on PE-1's CE-PE link
- Deactivate Quality of Service configuration on PE-1
- Deactivate rewrite-rules on Customer-A-Site-1's PE-CE link, thus placing all traffic into the Best_Effort class.

Configuration modifications required to enable Fast Reroute

- Enable CSPF on all virtual-routers in Service Provider core
- Activate path "Best_Path_cspf" and "Failover_Path_cspf" on PE-1 virtual router
- Activate Fast Reroute under MPLS configuration on PE-1 virtual router
- Activate Load Balancing Policy on PE-1
- Activate OSPF Traffic-Engineering extension on all virtual routers in the Service Provider core
- On P-2 and P-7 enable node-link-protection

3.3.7 Scenario 7: QoS Enabled and FRR Enabled – 8Mb of Traffic on a 10 Mb LSP

This scenario is useful to understand if any benefits are achieved when a virtualized Service Provider Core carries traffic on uncongested paths, i.e. carrying traffic at a lower bandwidth than its maximum bandwidth carrying capacity with Quality of Service being enabled. The link tends to have an additional 2 Mb of unused bandwidth. Round Trip Time can be used as a suitable metric, to illustrate the end to end latency. In addition to the above reason, I also wanted to understand how a virtualized service provider core would perform with Fast Reroute being enabled during failover from the primary path onto the secondary path. Junos RPM probe timeouts would help me understand the packet losses. The firewall filter enabled on Customer-A-Site-1's CE virtual router are "normal-dav" and "normal-voice". Below are the modifications made in the configuration for this scenario.

Configuration modifications required to enable Quality of Service

- Activate firewall filter "rpm-classifier" on PE-1's CE-PE link
- Activate Quality of Service configuration on PE-1
- Activate rewrite-rules on Customer-A-Site-1's PE-CE link, thus placing all traffic into the Best_Effort class.

Configuration modifications required to enable Fast Reroute

- Enable CSPF on all virtual-routers in Service Provider core
- Activate path "Best_Path_cspf" and "Failover_Path_cspf" on PE-1 virtual router
- Activate Fast Reroute under MPLS configuration on PE-1 virtual router
- Activate Load Balancing Policy on PE-1
- Activate OSPF Traffic-Engineering extension on all virtual routers in the Service Provider core
- On P-2 and P-7 enable node-link-protection

3.3.8 Scenario 8: QoS Enabled and FRR Enabled – 12Mb of Traffic on a 10 Mb LSP

This scenario is useful to understand how a virtualized Service Provider Core performs when the label switched path is congested, i.e. carrying traffic at its maximum bandwidth carrying capacity when Quality of Service has been enabled. Round Trip Time can be used as a suitable metric, to illustrate the end to end latency. In addition to the above reason, I also wanted to understand how a virtualized service provider core would perform with Fast Reroute being enabled during failover from the primary path onto the secondary path. Junos RPM probe timeouts would help me understand the packet losses. The firewall filter enabled on Customer-A-Site-1's CE virtual router are "congested-dav" and "congested-voice". Below are the modifications made in the configuration for this scenario.

Configuration modifications required to enable Quality of Service

- Activate firewall filter "rpm-classifier" on PE-1's CE-PE link
- Activate Quality of Service configuration on PE-1
- Activate rewrite-rules on Customer-A-Site-1's PE-CE link, thus placing all traffic into the Best_Effort class.

Configuration modifications required to enable Fast Reroute

- Enable CSPF on all virtual-routers in Service Provider core
- Activate path "Best_Path_cspf" and "Failover_Path_cspf" on PE-1 virtual router
- Activate Fast Reroute under MPLS configuration on PE-1 virtual router
- Activate Load Balancing Policy on PE-1
- Activate OSPF Traffic-Engineering extension on all virtual routers in the Service Provider core
- On P-2 and P-7 enable node-link-protection

3.4 Gathering and Analyzing the RPM Results

Based on the above scenarios, modifications were made to the configuration of the virtual routers. A new ssh session was opened with Customer-A-Site-1s CE router using the iTerm application in logging mode. This mode allows the contents displayed logged to a ".log" file. Next after running the test for approximately 300 seconds, the following operational mode command was given on Customer-A-Site-1's CE router, "show services rpm history-results | no-more". An abbreviated output of the same is visible in Figure 3.13. The output shows the name of the test probe being carried out along with the timestamp and Round Trip Time.

```
utkarsh@Customer-A-Site-1# run show services rpm history-results | no-more
  Owner, Test          Probe received              Round trip time
  Utkarsh, video_test     Fri Jan 13 02:53:47 2017      15989 usec
  Utkarsh, video_test     Fri Jan 13 02:53:48 2017      19832 usec
  Utkarsh, video_test     Fri Jan 13 02:53:49 2017      10582 usec
  Utkarsh, video_test     Fri Jan 13 02:53:50 2017      17442 usec
  Utkarsh, video_test     Fri Jan 13 02:53:51 2017      15836 usec
  Utkarsh, video_test     Fri Jan 13 02:53:52 2017      22300 usec
  Utkarsh, video_test     Fri Jan 13 02:53:53 2017      20580 usec
  Utkarsh, video_test     Fri Jan 13 02:53:54 2017      19397 usec
  Utkarsh, video_test     Fri Jan 13 02:53:55 2017      17573 usec
  Utkarsh, video_test     Fri Jan 13 02:53:56 2017      21245 usec
  Utkarsh, video_test     Fri Jan 13 02:53:57 2017      10449 usec
  Utkarsh, video_test     Fri Jan 13 02:53:58 2017      11140 usec
  Utkarsh, video_test     Fri Jan 13 02:53:59 2017      11173 usec
  Utkarsh, video_test     Fri Jan 13 02:54:00 2017      12526 usec
  Utkarsh, video_test     Fri Jan 13 02:54:01 2017      20160 usec
```

Figure 3.13: Abbreviated Output for RPM History Results

The output of the above command is logged automatically with the help of "iTerm" into a log file. Next, this log file is opened in the "TextEdit" application, where modifications are made to fill the blank spaces with "commas". Also the keyword "usec" has been removed for all entries using the "Find and Replace" option. Thereafter this file has been saved as a ".csv" extension file.

The benefit of doing this method, is I can simply open the ".csv" file into Microsoft Excel. Once the file is opened in Excel, the first two columns consisting of "Owner, Test" and "Probe received" are deleted and a new column is added prior to the "Round Trip Time" column which would have an incrementing timer from 1 second to 300 seconds called the "Seconds" column. Finally, the "Seconds" column and "Round-Trip Time" column are selected together and the following steps are performed to make the graph, Insert > Charts > X Y (Scatter) with Straight Lines. The "Seconds" columns forms the "X axis", whereas "Round Trip Time" forms the "Y axis".

3.5 Challenges

I faced a number of challenges when implementing the virtualized network. I consulted with my Thesis chair, committee members and industry experts to find solutions for the same. Some of the challenges faced are mentioned below:

- Generation of Voice Traffic: To emulate real-time voice traffic was a challenge. As I did not have access physical voice equipment and voice servers the only option left was to generate this traffic via software. Commercial grade tools are available which can generate voice traffic easily however due to financial costs this too was not an option. Next, I tried using SIPp and Asterisk which is an open source tool to generate SIP protocol traffic. Unfortunately, I couldn't get the setup up and running in time for both these applications. Thus, in order to generate voice traffic, I downloaded a sample SIP call with RTP in G711 from Wireshark's wiki page. Another challenge I faced is this sample call could not hit the incoming CE interface at the required speed. The workaround for this was to run multiple instances of Colasoft Packet Player in Burst Mode.

- Voice Traffic Reachability and Classification Issues: Another issue faced with voice traffic was that the sample packet downloaded from Wireshark's Wiki had destination address which were not part of the thesis test-bed. Due to this traffic could not reach the remote site. After some initial brainstorming, I decided to use the Tcprewrite application to change the destination mac address of all packets to a dedicated interface on Customer-A-Site-1's virtual router. This allowed traffic to be forwarded at Layer 2. Next, a default route was created on Customer-A-Site-2 and pushed to Customer-A-Site-1's virtual router with the help of the Service Provider. This allowed voice traffic for unknown destinations thru the Service Provider Core. The next challenge was to classify voice traffic when sent on the same interface as data and video traffic. No common parameters were found between successive voice traffic packets to be classified in a firewall filter. Hence, voice received its own dedicated interfaces. All traffic ingress on this interface would be categorized as voice traffic. Appropriate rewrite rules were configured to make sure traffic exits in a manner where PE-1 can classify it as voice.
- Scalability Limitations: My initial topology consisted of 11 "Provider" virtual routers. However, due to hardware limitations I had to reduce the number of "Provider" routers to 9.
- Unexpected Packet Loss and Fast Reroute feature check: When I initially configured the test bed and passed traffic thru core, I noticed that the failover scenarios led to 40 packet losses (FRR disabled). This value was too high. To understand if this was a performance constraint or normal behavior, I replicated the setup on physical Juniper Networks MX80 devices which my ex-employer was kind enough to let me test on. Even on the physical devices, I ran into 40 packets being lost. Next, I used IS-IS as the protocol to check if the same problem persists. 27 packet losses were visible when

IS-IS was used. After white-boarding the same, I realized it was the protocols default dead timers leading to packet drops. Configuring BFD aided in reducing the packet loss. Another challenge faced was Fast Reroute was not reducing the packet loss seen during failover scenarios. To confirm my configuration, I checked this feature on the MX80 devices and no packet loss was seen. This helped me clarify my doubt regarding FRR scalability in virtualized networks.

4 Simulation Results and Analysis

4.1 Scenario 1: QoS and FRR Disabled – 8 Mb of Traffic on a 10 Mb LSP

Scenario 1 compares three different types of traffic, namely voice, video and data traffic on a non-congested path i.e. 8 Mb of traffic on a 10 Mb label switched path. In this scenario, tests were carried out with Quality of Service and Fast Reroute being disabled.

4.1.1 Voice Traffic

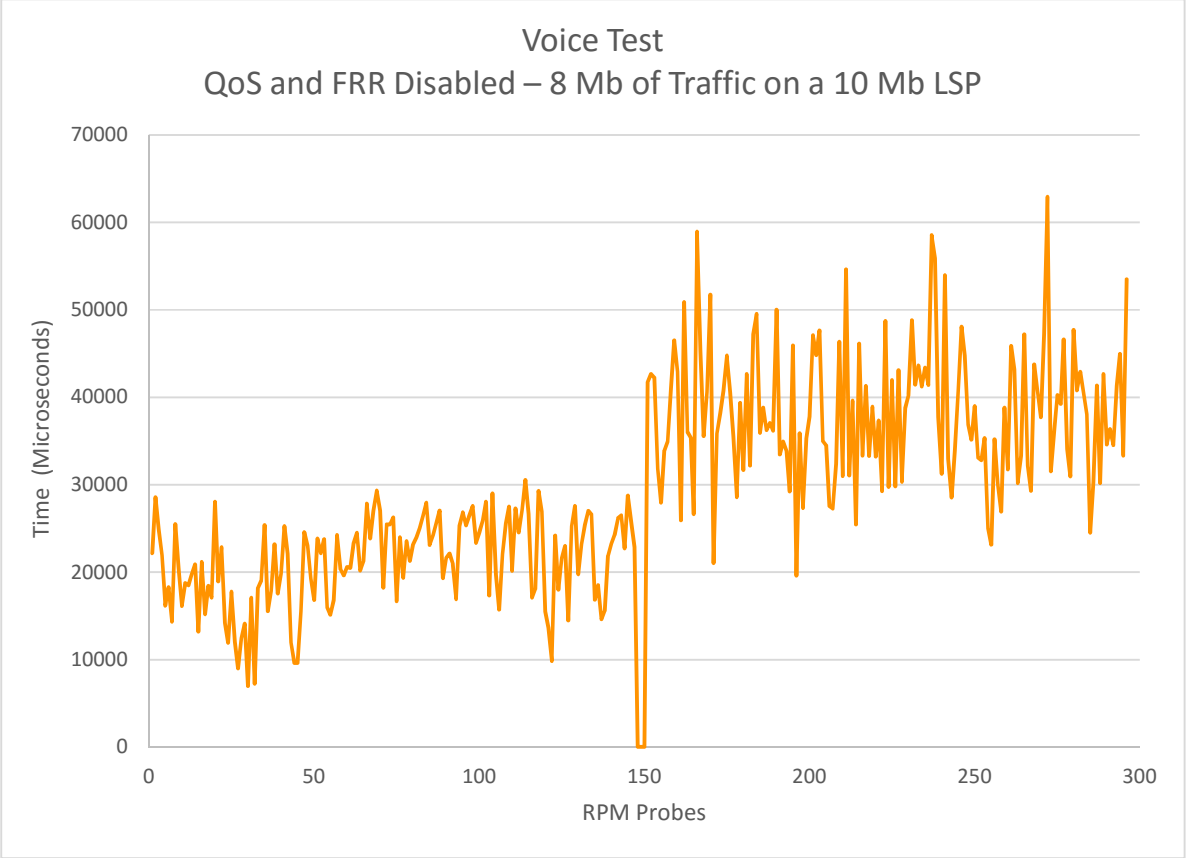

Figure 4.1: Performance of Voice Traffic - QoS and FRR Disabled – 8 Mb of Traffic on a 10 Mb LSP

Within this test, voice traffic along with video and data traffic is classified as best effort traffic, since QoS has been disabled. Three different types of traffic fight for the

bandwidth on the link. As we can see from the graph above, when RPM probes are sent alongside voice traffic, the maximum RTT value reached is 30556 microseconds on the primary label switched path. At approximately 150 seconds a failover occurs, redirecting traffic onto the secondary path. While on the secondary path the highest RTT value reached is 62947 microseconds. A spike in RTT values is clearly visible from the above graph, post failover.

During the failover, we can see that three RPM probes have timed out. The three RPM probes which have timed out can be attributed to the BFD protocol, whose timers take approximately three seconds to detect a link failure. Thus, three successive packets are dropped in this scenario until MPLS finally moves traffic over a secondary path.

When compared against the performance of Voice traffic in Scenario 3 (8 Mb of Traffic with QoS Enabled) we see that having quality of service enabled does give us a minor advantage by keeping the round-trip times shorter than quality of service being disabled. This is more evident for traffic on the secondary path. This shows us that in a virtualized service provider environment, service providers can still send voice traffic without prioritizing this traffic, the only trade-off being a higher end to end latency when compared against QoS enabled scenarios. This could be suitable in certain cases for free customers.

What is also noticeable when comparing the above scenario against voice traffic in Scenario 3 is the minimum RTT values which are slightly lower in Scenario 1. The lowest RTT recorded in in Scenario 1 is 6953 microseconds, whereas in Scenario 3 (QoS Enabled), is 10895 microseconds. Although this value may seem trivial, it appears that the processing time taken by the virtual router to put the voice traffic into the Voice Forwarding Class leads to this delay.

4.1.2 Video Traffic

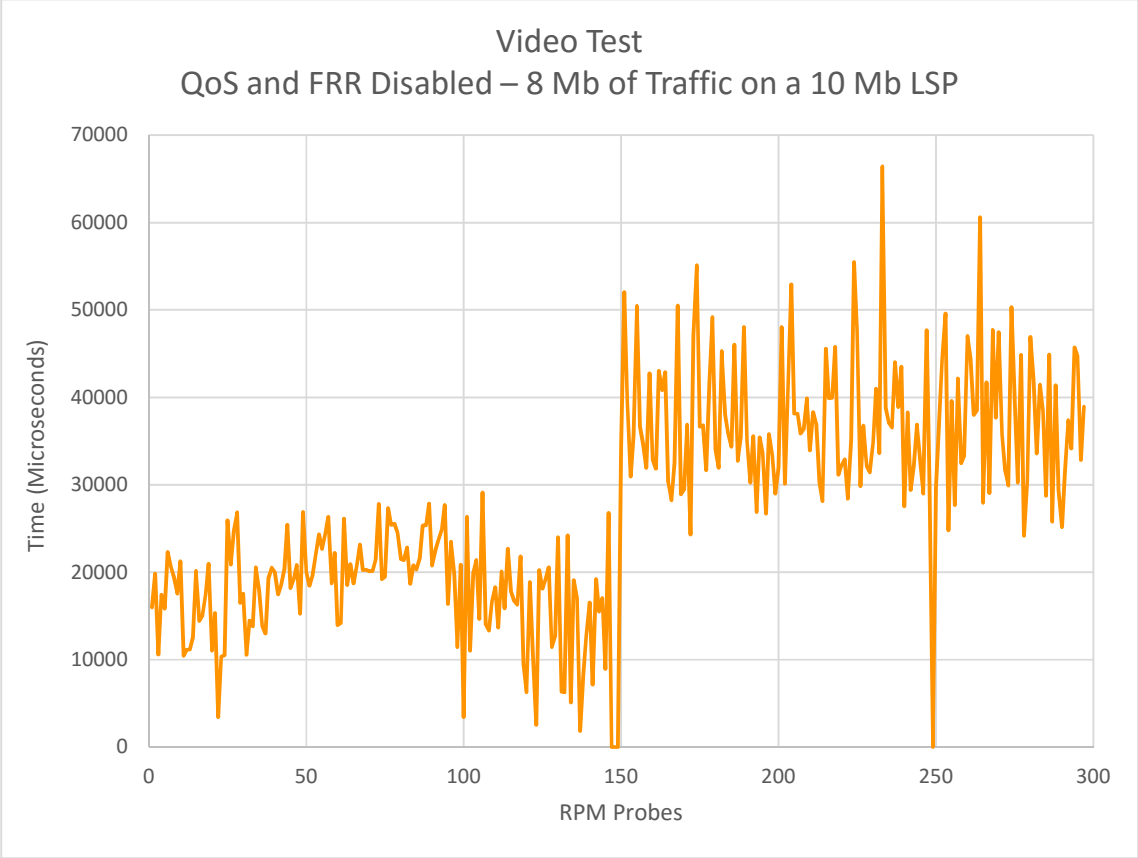

Figure 4.2: Performance of Video Traffic - QoS and FRR Disabled – 8 Mb of Traffic on a 10 Mb LSP

In scenario 1, video traffic is mixed alongside data and voice into the best effort forwarding class. This is due to the fact that QoS has been disabled. When Video traffic flows on the primary path, we can see low RTT values with zero packet drops. The low RTT values are a result of the small packet sizes being received and directly placed on the wire due to QoS being disabled. When compared against Scenario 3 which has QoS enabled we observe a few microseconds being saved when QoS has been disabled. At approximately 150 seconds a failover is initiated, leading to three packet losses. Unfortunately, post failover when traffic is redirected over a secondary link we can see the spike in RTT values, which

are almost double than when on the primary path. The trend on the graph also shows us a few packet losses for video traffic being sent over the secondary path.

4.1.3 Data Traffic

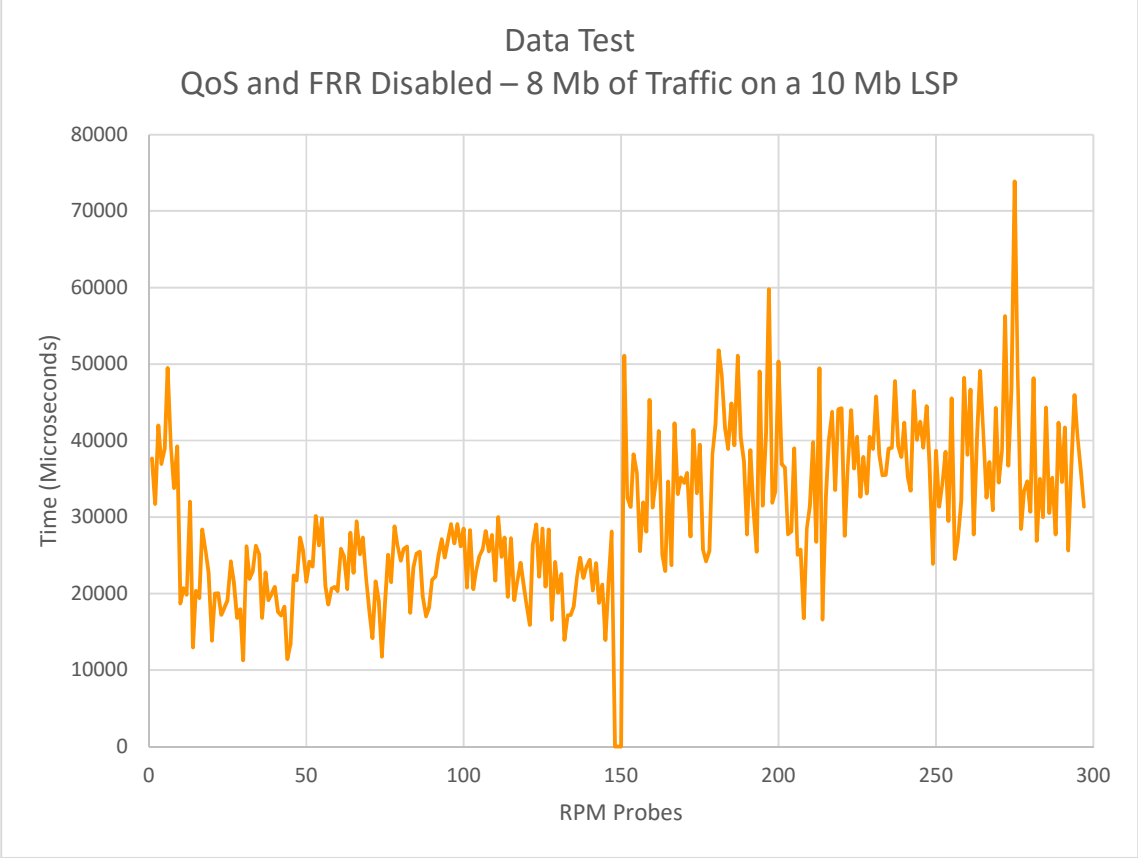

Figure 4.3: Performance of Data Traffic - QoS and FRR Disabled – 8 Mb of Traffic on a 10 Mb LSP

In the above graph, we can see the performance of data traffic when QoS has been disabled and traffic is flowing at less than the paths maximum bandwidth carrying capacity. The trend shows us that since the path can handle the current traffic load, there are no packet losses seen, except during the time of failover. We can see an initial spike in traffic on the primary path, before actually stabilizing between 11000 to 30000 microseconds. Post failover

we can see several spikes of traffic, almost doubling the RTT values when on the secondary path.

Whilst comparing the same scenario with the scenario 3, where QoS has been enabled we can see that QoS tends to reduce the end to end delay (lower is better), with the help of Packet Scheduling and Queue Prioritization. We observe approximately a 1.5 time increase in Round Trip Time when traffic is sent on the primary path when Quality of Service has been disabled. Based on the data application being used, TCP could resend the lost packets seen during failover.

4.2 Scenario 2: QoS and FRR Disabled – 12 Mb of Traffic on a 10 Mb LSP

Scenario 2 compares three different types of traffic, namely voice, video and data traffic on a congested path i.e. 12 Mb of traffic on a 10 Mb label switched path. In this scenario, tests were carried out with Quality of Service and Fast Reroute being disabled..

4.2.1 Voice Traffic

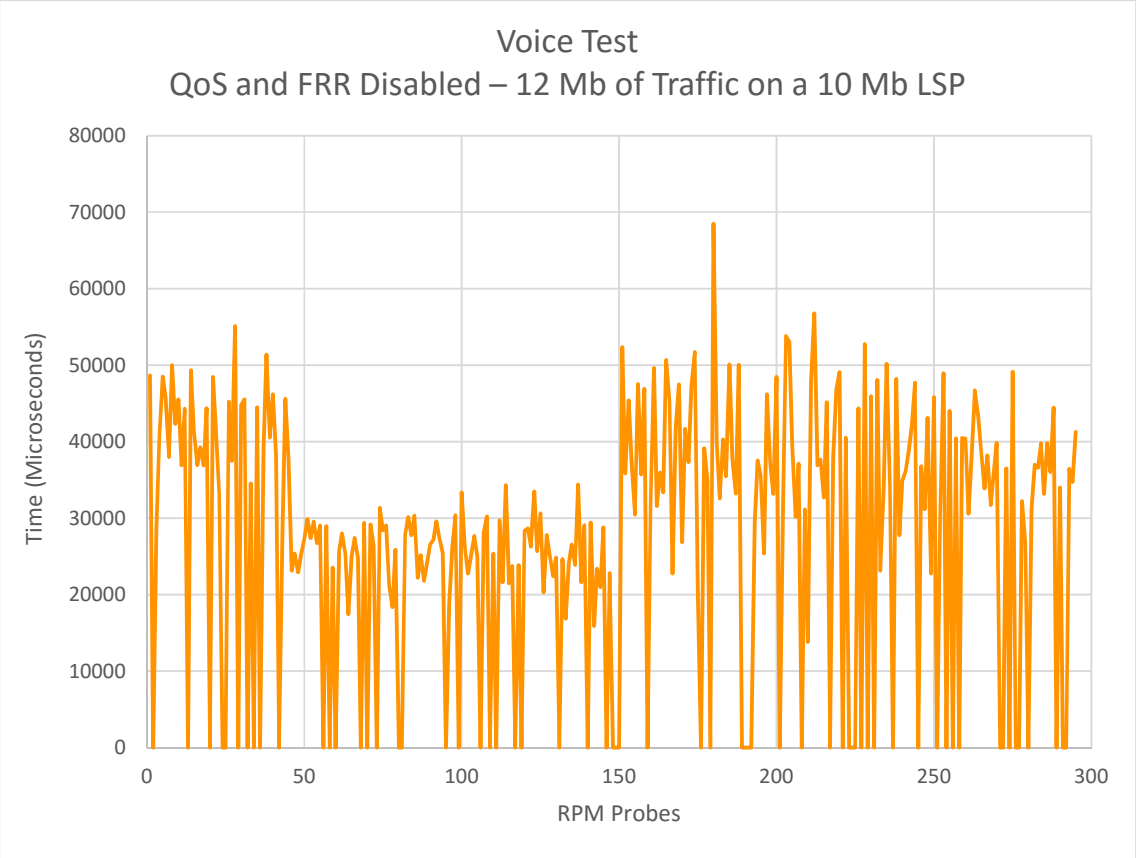

Figure 4.4: Voice Traffic - QoS and FRR Disabled – 12 Mb of Traffic on a 10 Mb LSP

Since QoS has been disabled, voice traffic along with video and data traffic is classified as best effort traffic. Three different types of traffic fight for the bandwidth on the link. The graph above depicts the performance of voice traffic over a congested path, wherein the path receives more traffic than it is meant to handle. As we can see, the initial RTT's are

significantly high and excessive RTT probe timeouts occurs, citing a failure of RTT replies from the peer endpoint. The highest RTT value recorded on the primary path is 55118 microseconds. At approximately 150 seconds, a failover is initiated resulting in three packet losses which draws it affiliation to BFD's timers. What is surprising to see from this scenario is, even in periods of peak congestion the virtual routers internal mechanism for handling packet switchovers drops only three packets. This result is similar to traffic drops seen during failovers on uncongested links. From the 295 RTT probes sent, the virtual routers in the service provider core have only dropped 63 probe as compared to Scenario 4 (QoS Enabled – 12 Mb traffic on a 10 Mb LSP) which dropped 75 probes (lower is better). What this does show us is that although having QoS disabled pushes more traffic thru the service provider core the drawback are having higher RTT values on the primary and secondary path. This point can be further emphasized when comparing Scenario 2 and Scenario 4's voice traffic.

Scenario 2 (QoS Disabled), while allowing more traffic has an initial higher RTT almost crossing 50000 microseconds until finally averaging at approximately 31000 microseconds on the primary path. Whereas in Scenario 4 where QoS is enabled, we can see that when Voice buffers fill up, more packets are dropped. In scenarios with peak congestion we see that having Quality of Service enabled does not add a significant advantage for traffic sent across the secondary path.

4.2.2 Video Traffic

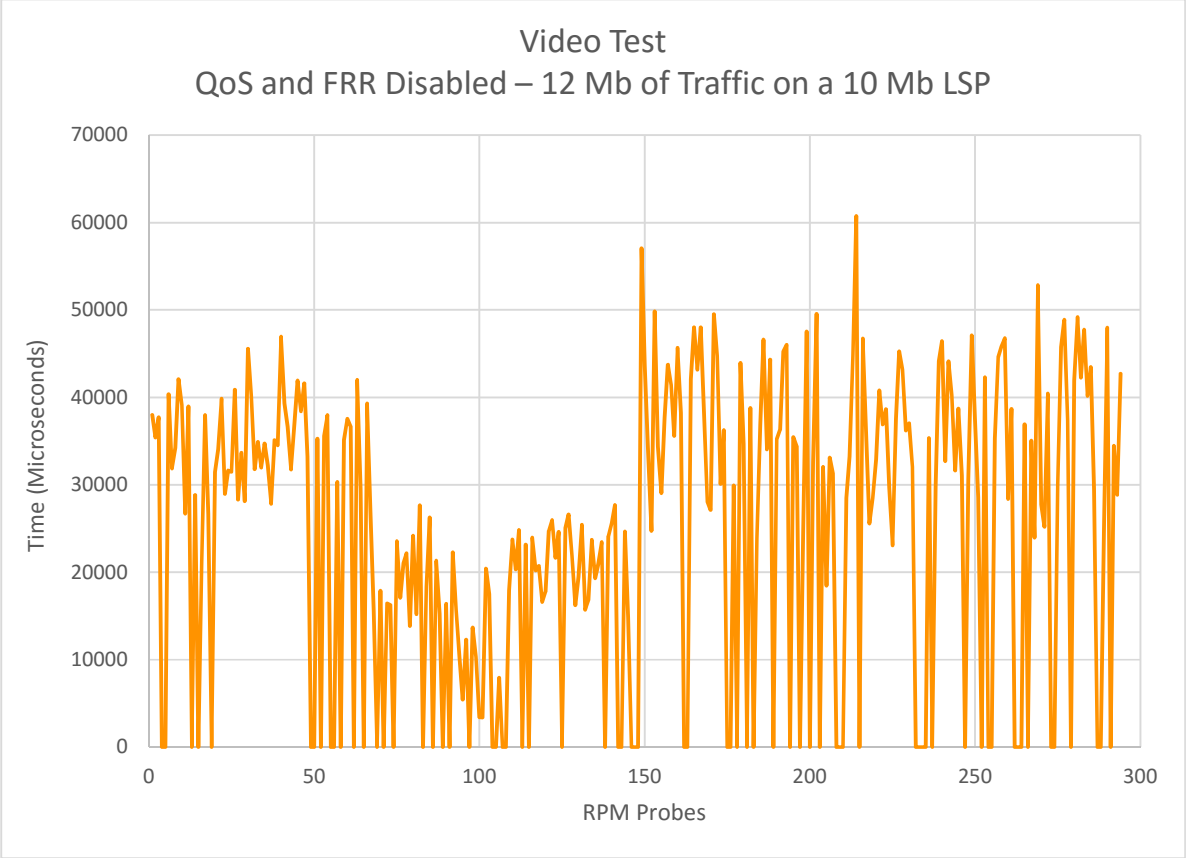

Figure 4.5: Performance of Video Traffic - QoS and FRR Disabled – 12 Mb of Traffic on a 10 Mb LSP

In Scenario 2, three different types of traffic are pushed at more than the paths bandwidth carrying capacity and are merged together due to the lack of QoS. What can be seen in this scenario as compared to Scenario 4 (QoS Enabled) is three lesser RTT probe losses for three hundred RPM probes. Although this may seem trivial it still shows that QoS does not play a huge role in improving congestion in a virtualized network to reduce end to end latency for video traffic on a congested path. What is also noticeable is the significant increase in the end-to-end delay as compared with Scenario 1, where 8 Mb of traffic flows thru a QoS disabled network. The reason for this behavior stems from the fact that due to link

saturation, not enough packets can flow thru the network. This results in high RTT values and increased probe losses.

4.2.3 Data Traffic

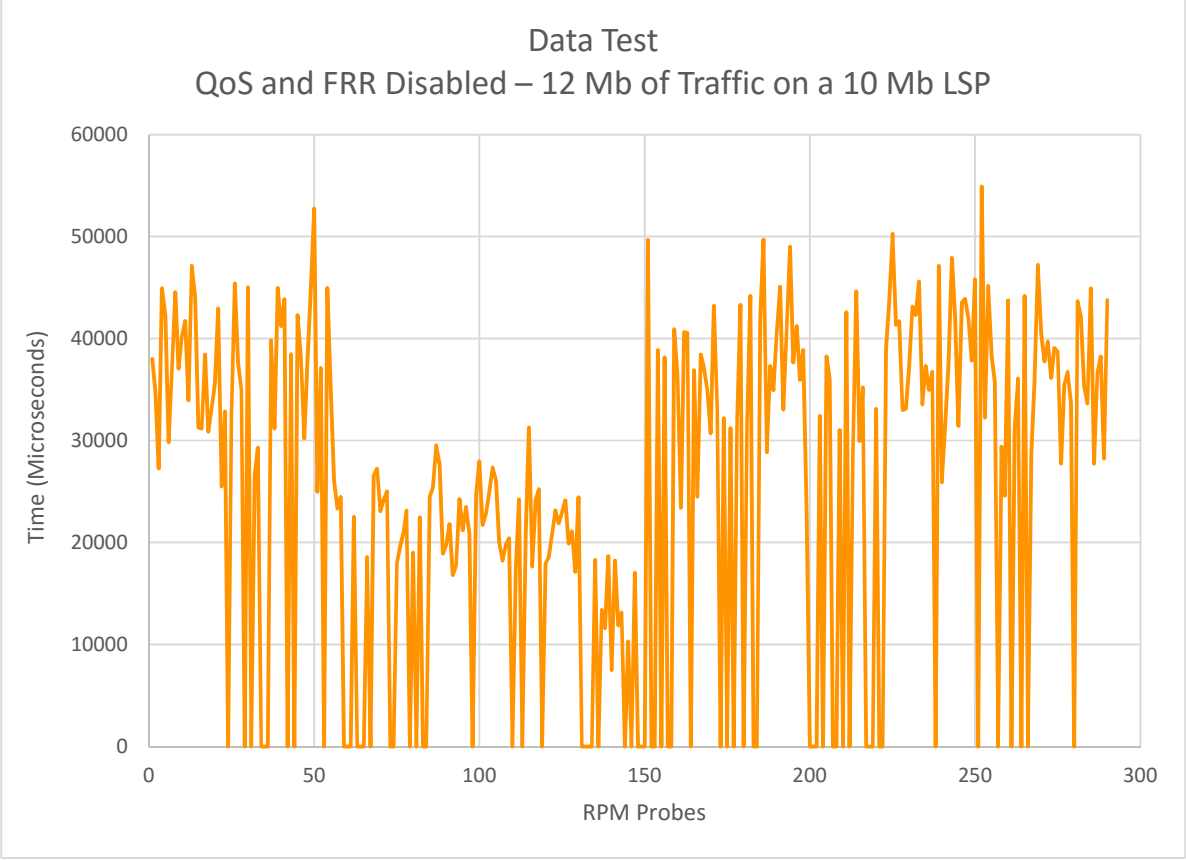

Figure 4.6: Performance of Data Traffic - QoS and FRR Disabled – 12 Mb of Traffic on a 10 Mb LSP

When comparing the above graph against Scenario 4's data traffic we can observe that having Quality of Service disabled lets more data traffic be sent across the service provider core. However, the same comes at the cost of affecting the other traffic, more specifically voice. This is due to the size of the data packets used in the tests which are approximately 1500 bytes compared to voice which are approximately only 450 bytes. We can see that having QoS disabled leads to an increase in the overall Round Trip Time as compared to

when QoS has been enabled. Another striking observation that can be made from the above graph is, data traffic tends to have consistently high Round Trip Times irrespective of whether it is flowing on the primary path or secondary path.

4.3 Scenario 3: QoS Enabled and FRR Disabled – 8Mb of Traffic on a 10 Mb LSP

Scenario 3 compares three different types of traffic, namely voice, video and data traffic on a non-congested path i.e. 8 Mb of traffic on a 10 Mb label switched path. In this scenario, tests were carried out with Quality of Service being enabled and Fast Reroute being disabled.

4.3.1 Voice Traffic

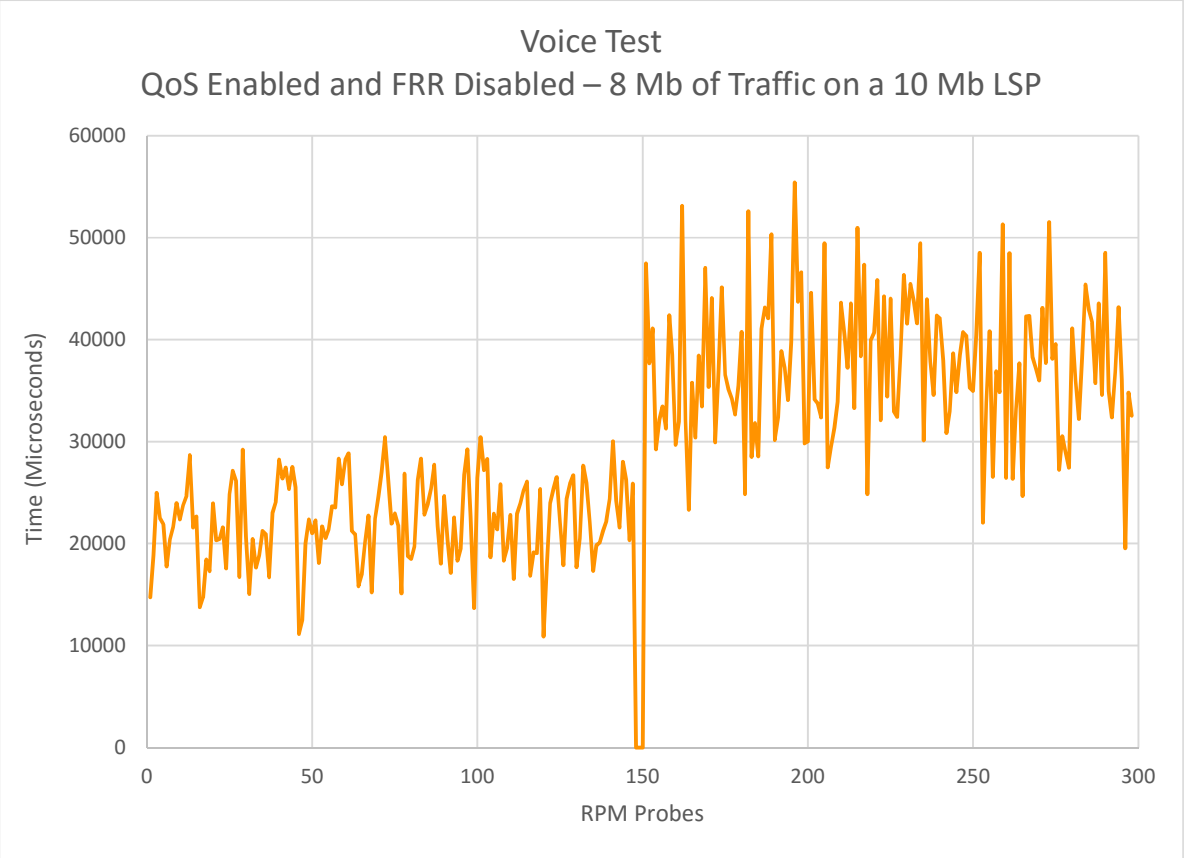

Figure 4.7: Performance of Voice Traffic - QoS Enabled and FRR Disabled – 8 Mb of Traffic on a 10 Mb LSP

As you can see from the graph above, when RPM probes are sent alongside voice traffic, the maximum RTT value reached is 30045 microseconds on the primary path. At approximately 150 seconds a failover occurs, redirecting traffic onto the secondary path.

While on the secondary path the highest RTT value reached is 55408 microseconds. A spike in RTT values is clearly visible from the above graph, post failover depicting longer end to end latency when traffic flows on the secondary paths.

During the failover, we can see that three RPM probes have timed out. The three RPM probes which have timed out can be attributed to BFDs timers which takes approximately 3 seconds to detect a link failure. When comparing scenario 1 with Scenario 3, the latter having QoS enabled with both sending 8 Mb of traffic on a 10 Mb LSP, shows us that QoS does stand to have benefits by reducing the Round Trip Time/end-to-end latency. The maximum RTT reported on a primary path in Scenario 1 is 30556 microseconds whereas in Scenario 3, with QoS enabled is reduced to a maximum of 30449 microseconds (lower is better). Although this difference may seem small, it still sheds light that QoS does have its benefits when used even on uncongested links. This shows us that in a virtualized service provider environment, service providers can use QoS to prioritize voice traffic and reduce the end-to-end delay.

What is also noticeable when comparing the above scenario against voice traffic in Scenario 1, are the minimum RTT values which are slightly higher than Scenario 1. The lowest RTT recorded in in Scenario 1 is 6953 microseconds, whereas in Scenario 3 (QoS Enabled), is 10895 microseconds. According to me, an observation can be made where the processing delay taken by the virtual router to classify the voice traffic into the Voice Forwarding Class and back onto the wire leads to this small delay.

4.3.2 Video Traffic

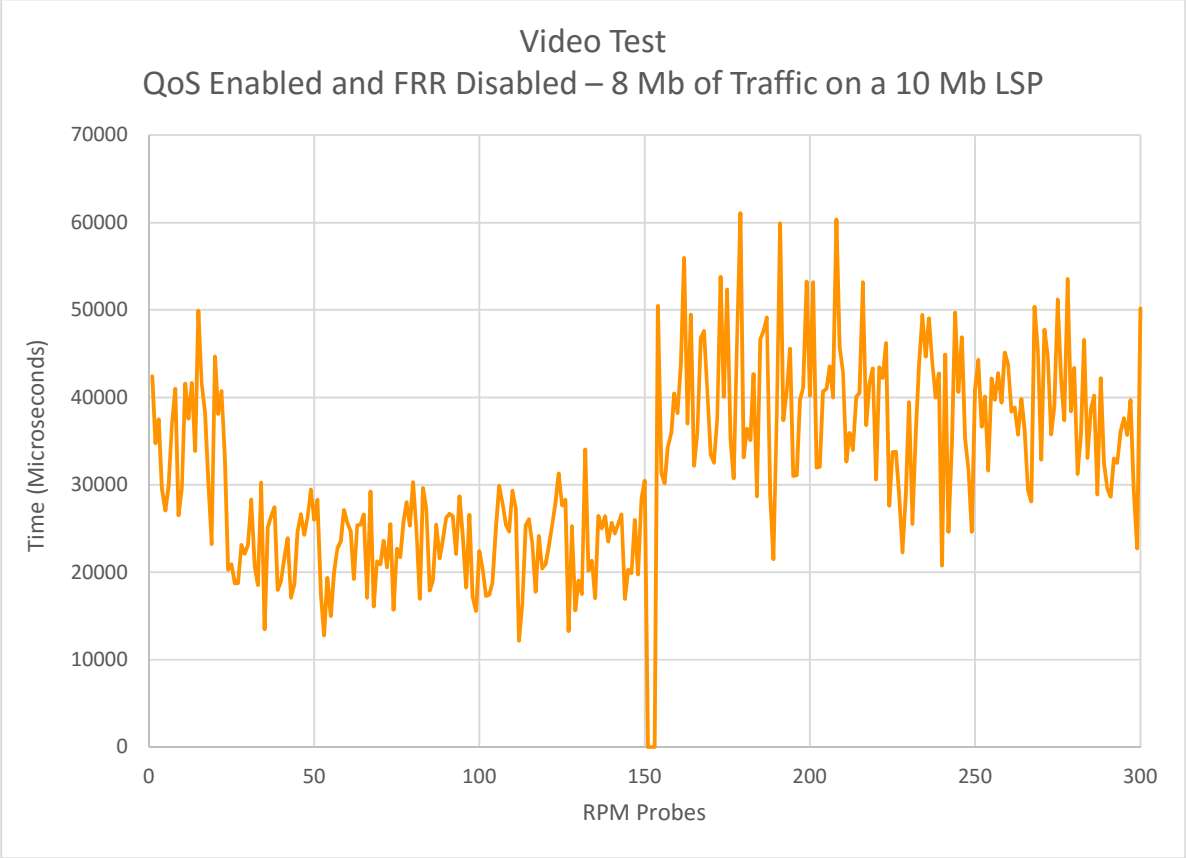

Figure 4.8: Performance of Video Traffic - QoS Enabled and FRR Disabled – 8 Mb of Traffic on a 10 Mb LSP

In scenario 3, the three types of traffic each have separate queues for Quality of Service. We can see that initially the traffic peaks higher than Scenario 1 wherein QoS is disabled for approximately 35 RTT probes, following which we can see a steady graph in which the RTT values lie between 20000 to 30000 microseconds. When a failover is initiated, only three RTT probes are lost, which again accounts to the BFD protocols timers. Post failover, we can see a small improvement in reducing the end to end delay within the network, showing us that QoS does offer a small benefit for video traffic when being routed across secondary paths. The graph also shows us how QoS handles congestion, by buffering

packets and preventing any packet drops on both the paths. The graph also shows us that as long as the link is carrying traffic below its maximum bandwidth carrying capacity, the scheduling mechanisms handling Quality of Service prevent packet loss.

4.3.3 Data Traffic

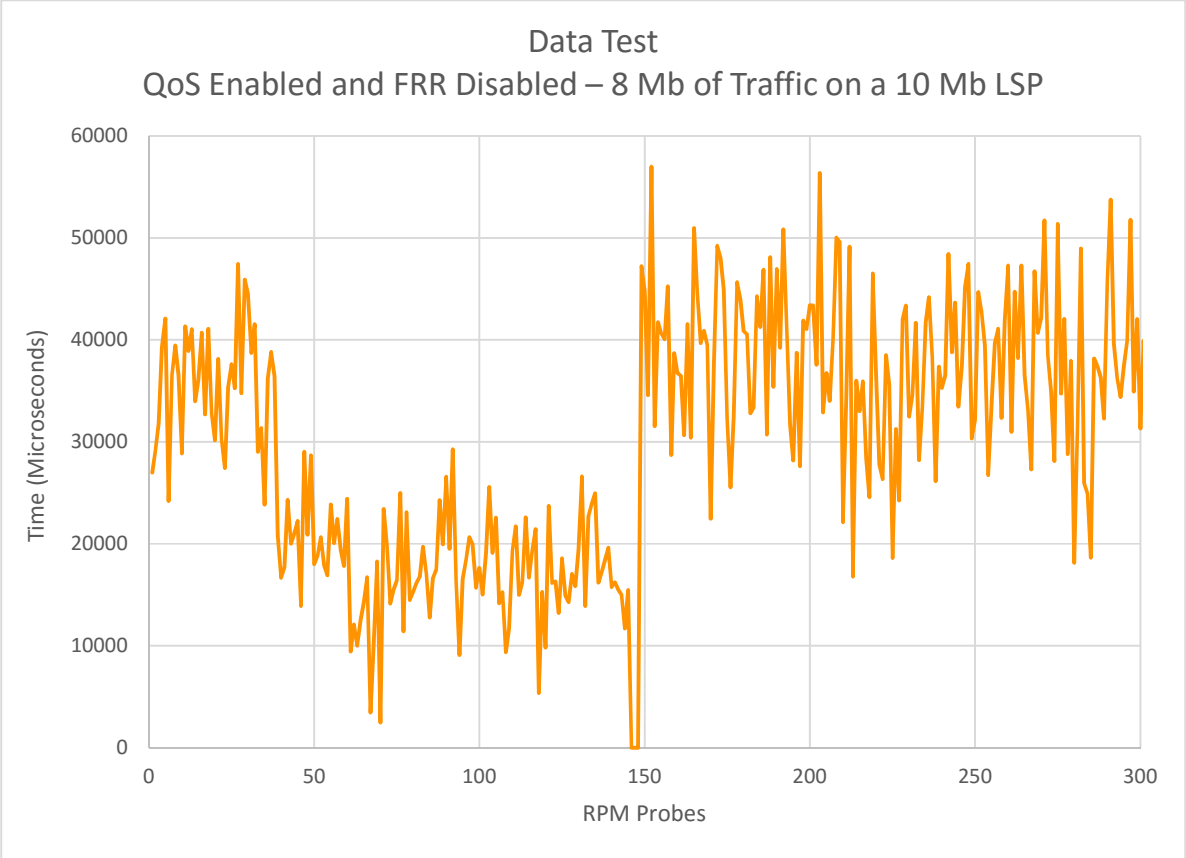

Figure 4.9: Performance of Data Traffic - QoS Enabled and FRR Disabled – 8 Mb of Traffic on a 10 Mb LSP

In the above graph we can see the performance of data traffic when QoS has been enabled and traffic is flowing at less than maximum path carrying capacity. The trend shows us that since the path can handle the current traffic load there are no packet losses seen,

except during the time of failover. We can see an initial spike of traffic on the primary path, before actually stabilizing between 10000 to 20000 microseconds. Post failover we can see several spikes of traffic, almost doubling the RTT values when on the secondary path. This behavior is normal as the secondary path has additional hops.

Whilst comparing the same scenario, with the Scenario, where QoS is disabled we can clearly see that QoS tends to reduce the end to end delay (lower is better), with the help of Packet Scheduling and Queue Prioritization. QoS has helped us by trying to reduce packet bursts from increasing the Round-Trip Times. Overall the performance of data traffic seems to improve when QoS has been applied in a virtualized network and traffic is well below the paths maximum bandwidth carrying capacity. Based on the data application being used, TCP could resend the lost packets seen during failover.

4.4 Scenario 4: QoS Enabled and FRR Disabled – 12Mb of Traffic on a 10 Mb LSP

Scenario 4 compares three different types of traffic, namely voice, video and data traffic on a congested path i.e. 12 Mb of traffic on a 10 Mb label switched path. In this scenario, tests were carried out with Quality of Service being enabled and Fast Reroute being disabled.

4.4.1 Voice Traffic

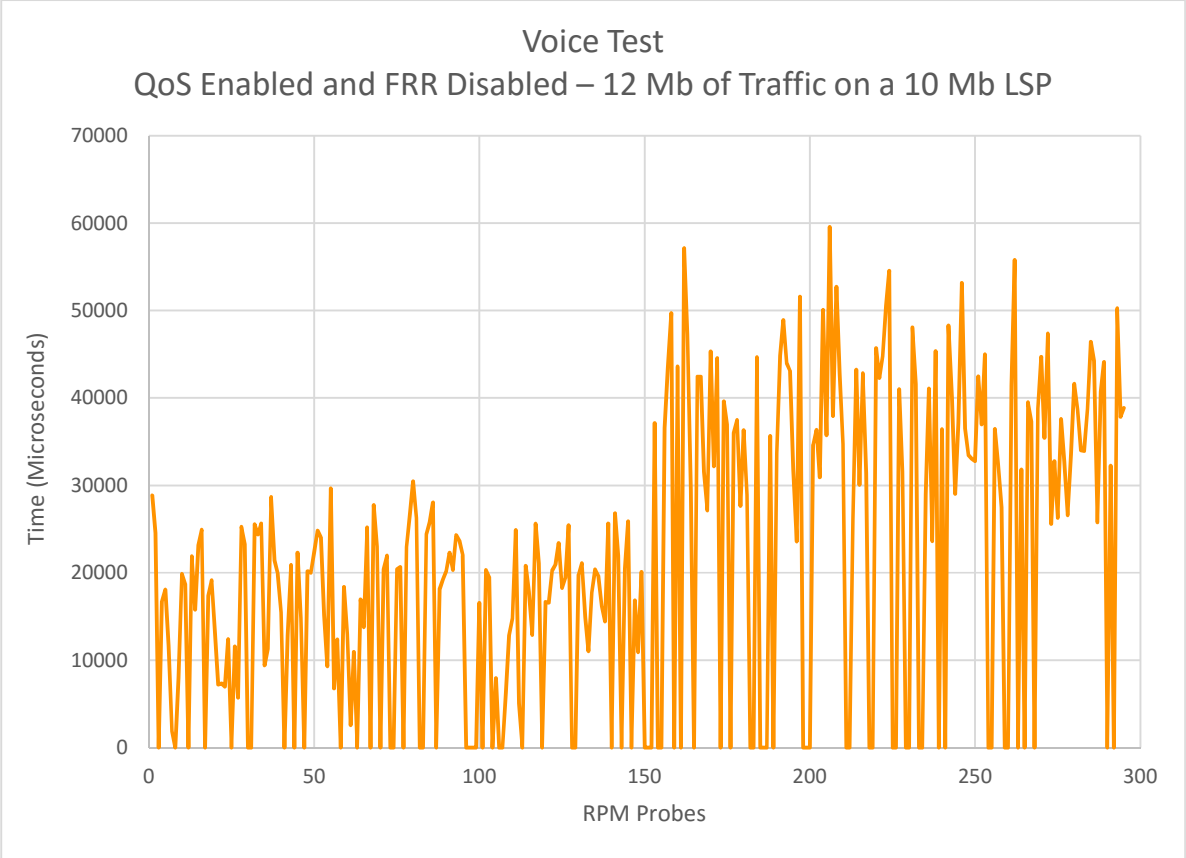

Figure 4.10: Performance of Voice Traffic – QoS Enabled and FRR Disabled – 12 Mb of Traffic on a 10 Mb LSP

Three different types of traffic fight for the bandwidth on the link, however due to QoS being enabled each traffic reserves a certain amount of bandwidth on the link. The graph above depicts the performance of voice traffic over a congested path, wherein the path

receives more traffic than it is meant to handle. The highest RTT value recorded on the primary path is just a little over 30000 microseconds. At approximately 150 seconds, a failover is initiated resulting in three packet losses which draws it affiliation to BFD's timers. Keeping in mind, QoS has been enabled we can see the routers end goal is not to allow the most number of packets rather to try and reduce the end-to-end delay. To illustrate this point further, the virtual routers in the service provider core have dropped approximately 75 probe as compared to Scenario 2 (QoS Disabled – 12 Mb traffic on a 10 Mb LSP) which dropped 63 probes. The benefits of QoS seem to simmer down when traffic is routed over a secondary path as both the scenarios, Scenario 2 (QoS disabled) and Scenario 4 show similar results for voice traffic being sent across the secondary path. Thus, the benefits of QoS on a congested link in a virtualized service provider core are only seen across the primary path. This observation can be seen when voice traffic is compared against Scenario 2's voice traffic. Average Round-Trip Times tends be 2x times higher than seen in Scenario 4.

4.4.2 Video Traffic

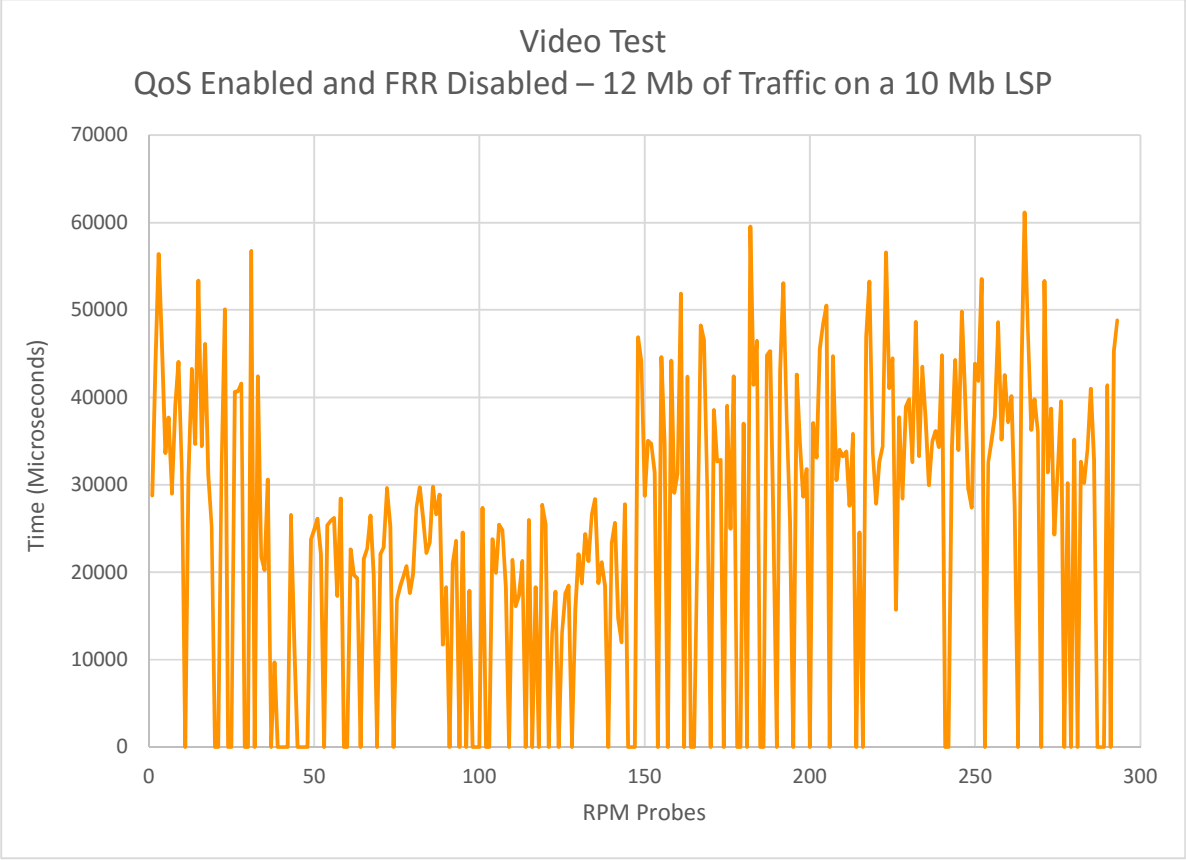

Figure 4.11: Performance of Video Traffic – QoS Enabled and FRR Disabled – 12 Mb of Traffic on a 10 Mb LSP

In Scenario 4, QoS has been enabled with voice, video and data traffic each receiving a dedicated amount of bandwidth. What we do notice in this scenario is, QoS fails to reduce the end to end RTT significantly when compared to scenarios where no QoS is applied. However, it does improve the overall performance of the other types of traffic flowing thru the network. This is done by not starving other traffic queues. As quality of service has been enabled, we can see that whenever video traffic exceeds its committed information rate, Quality of Service starts policing the excess traffic. When compared to scenarios wherein only 8 Mb of traffic is sent on the path we can see a significantly higher number of Round-

Trip Timeouts showing us that due to congestion the end-to-end latency is increasing and as traffic continues to burst more packets are being dropped.

4.4.3 Data Traffic

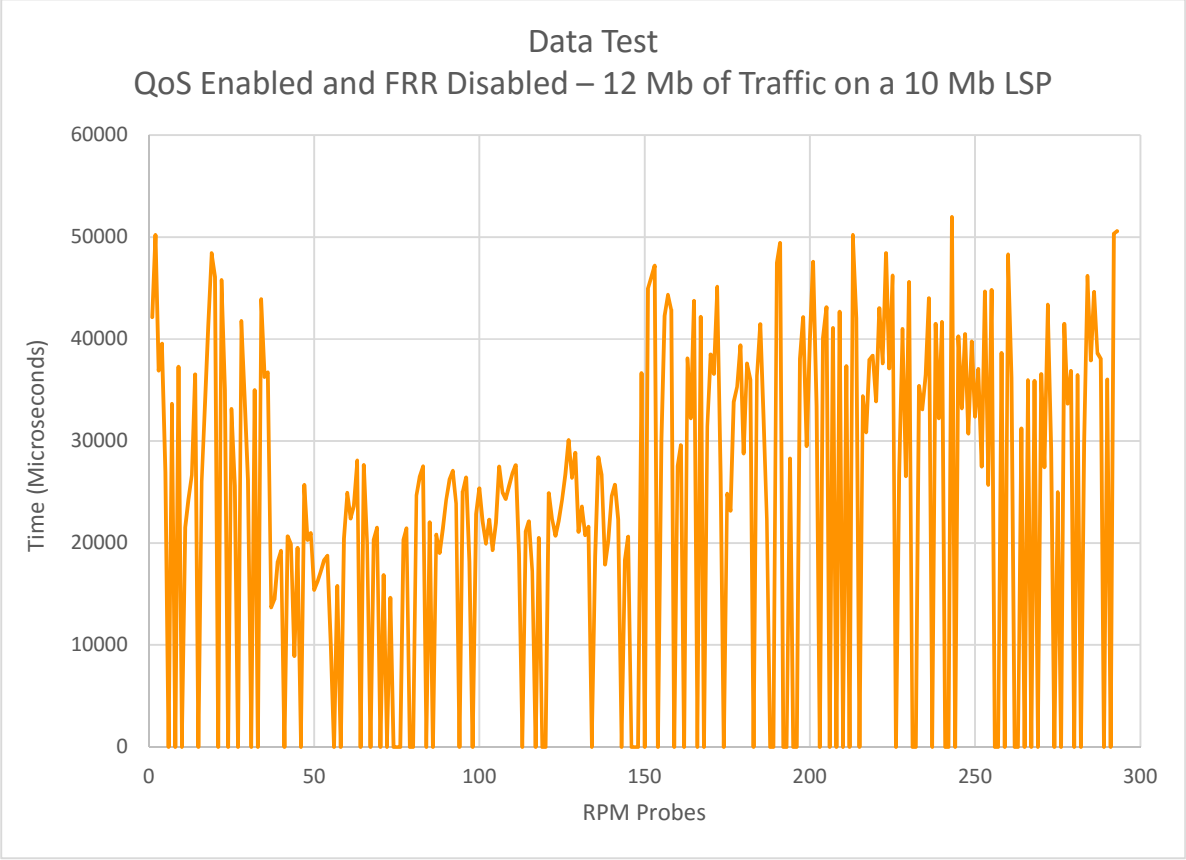

Figure 4.12: Performance of Data Traffic – QoS Enabled and FRR Disabled – 12 Mb of Traffic on a 10 Mb LSP

In the above graph, we can see the performance of Data traffic being on a congested path. Since Quality of Service has been enabled we can see a high number of packet drops. As data traffic has the highest payload size we see the buffer filling up more quickly than voice and video. When comparing Scenario 4's data traffic test with Scenario 2's data traffic test for data traffic flowing over the primary path we see that since Quality of Service has been enabled and data traffic is violating its maximum transmit speed, Quality of Service tends to police the violating traffic. When comparing Scenario 4's data traffic test with Scenario 2's data traffic test, we see similar results over the secondary path. It would be

interesting to study if we can reduce packet loss with Quality of Service enabled by changing the buffer sizes. This has been mentioned in the future section of this thesis.

4.5 Scenario 5: QoS Disabled and FRR Enabled – 8Mb of Traffic on a 10 Mb LSP

Scenario 5 compares three different types of traffic, namely voice, video and data traffic on a non-congested path i.e. 8 Mb of traffic on a 10 Mb label switched path. In this scenario, tests were carried out with Quality of Service being disabled and Fast Reroute being enabled.

4.5.1 Voice Traffic

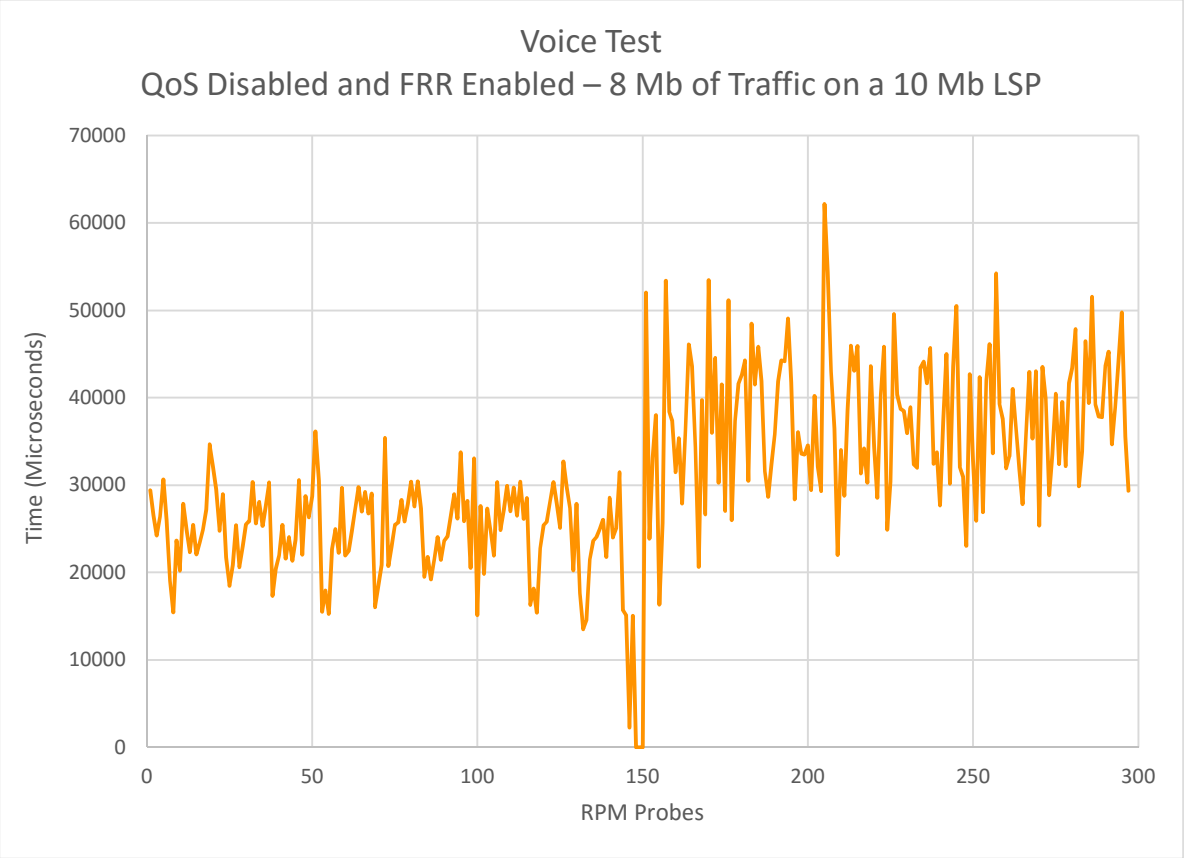

Figure 4.13: Performance of Voice Traffic - QoS Disabled and FRR Enabled – 8 Mb of Traffic on a 10 Mb LSP

Scenario 5's results tend to show similar results as seen in Scenario 1. With QoS being disabled all traffic types share the queues (best effort) properties. Traffic seems to be steady on the primary path between 15000 to 36000 microseconds. Post failover, we can see the spike in the graph when traffic moves over the secondary path. The reason for the spike in RTT, stems from the fact the increased hop counts to reach the egress node. When comparing Scenario 5 with Scenario 7, we see the difference in performance when having QoS disabled vs having QoS enabled. Having QoS disabled we can see higher average RTT values than when QoS is enabled. An observation for this slight increase in RTT values could be due to voice, video and data traffic being merged into best effort traffic class.

What is unexpected to see, is that having Fast Reroute enabled in a virtualized service provider core, does not offer any significant benefit in reducing packet loss for voice traffic. As compared with Scenario 1, where Fast Reroute has been disabled we still see three RTT probe timeouts. The reason we see three RTT probe timeouts and not forty probe timeouts is due to the fact, Bidirectional Forwarding Detection has been enabled for the underlying IGP, whose timers detect a link failure within three seconds.

4.5.2 Video Traffic

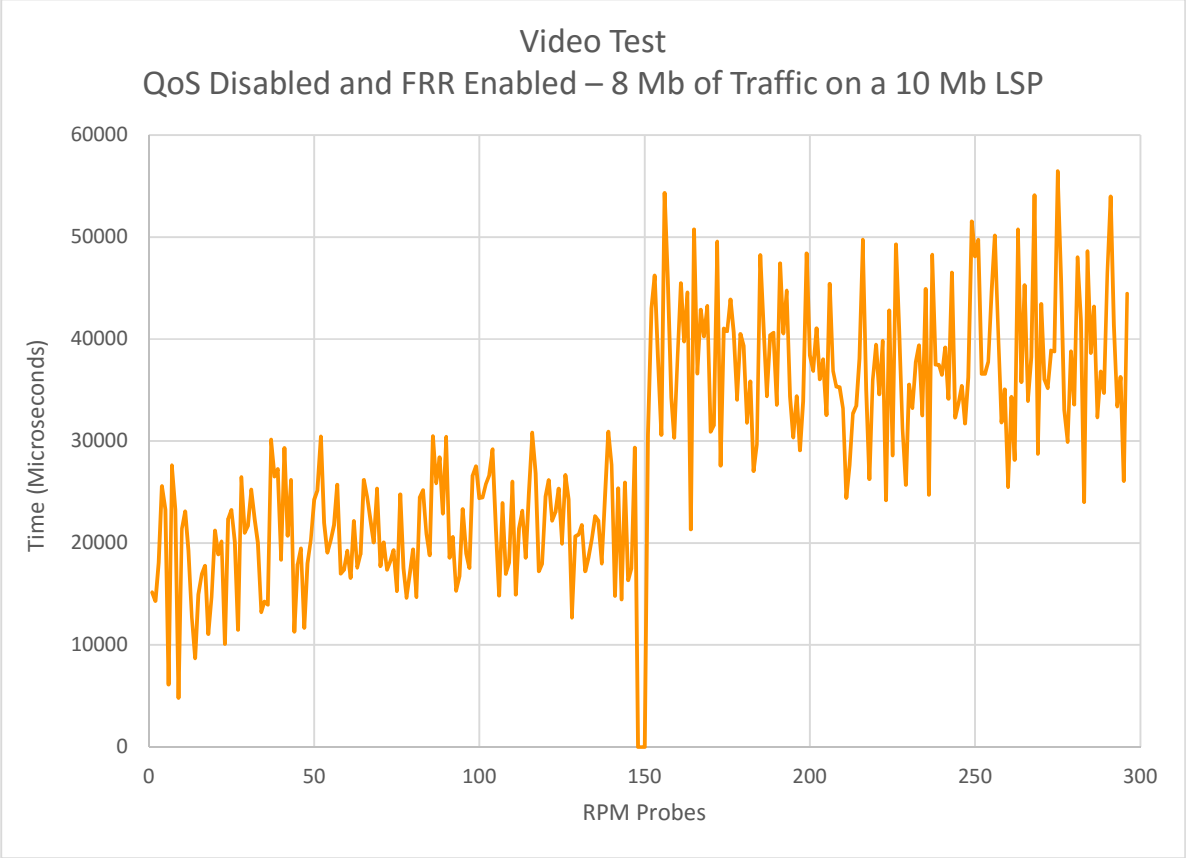

Figure 4.14: Performance of Video Traffic - QoS Disabled and FRR Enabled – 8 Mb of Traffic on a 10 Mb LSP

In Scenario 5, we can see the performance of video traffic when QoS has been disabled but Fast Reroute has been enabled. Video traffic tends to show low RTT values when being sent on the primary path. The reason of such low RTT values, being shown even though QoS has been disabled appears to be due to the small packet sizes being sent on a non-congested path. We can see that during failover, even though Fast Reroute has been enabled, we see three round trip time probe timeouts. The results of this test seem to be similar to video traffic tests carried out in Scenario 1. It appears even when Quality of Service has been disabled and the link is carrying traffic lower than its maximum bandwidth

carrying capacity, the fast reroute mechanism cannot alleviate the three-packet loss seen during failover. As mentioned in the future section of this thesis, it would be interesting to see if resiliency features are improved when tested on different hypervisors and configuring additional resources to the virtual routers.

4.5.3 Data Traffic

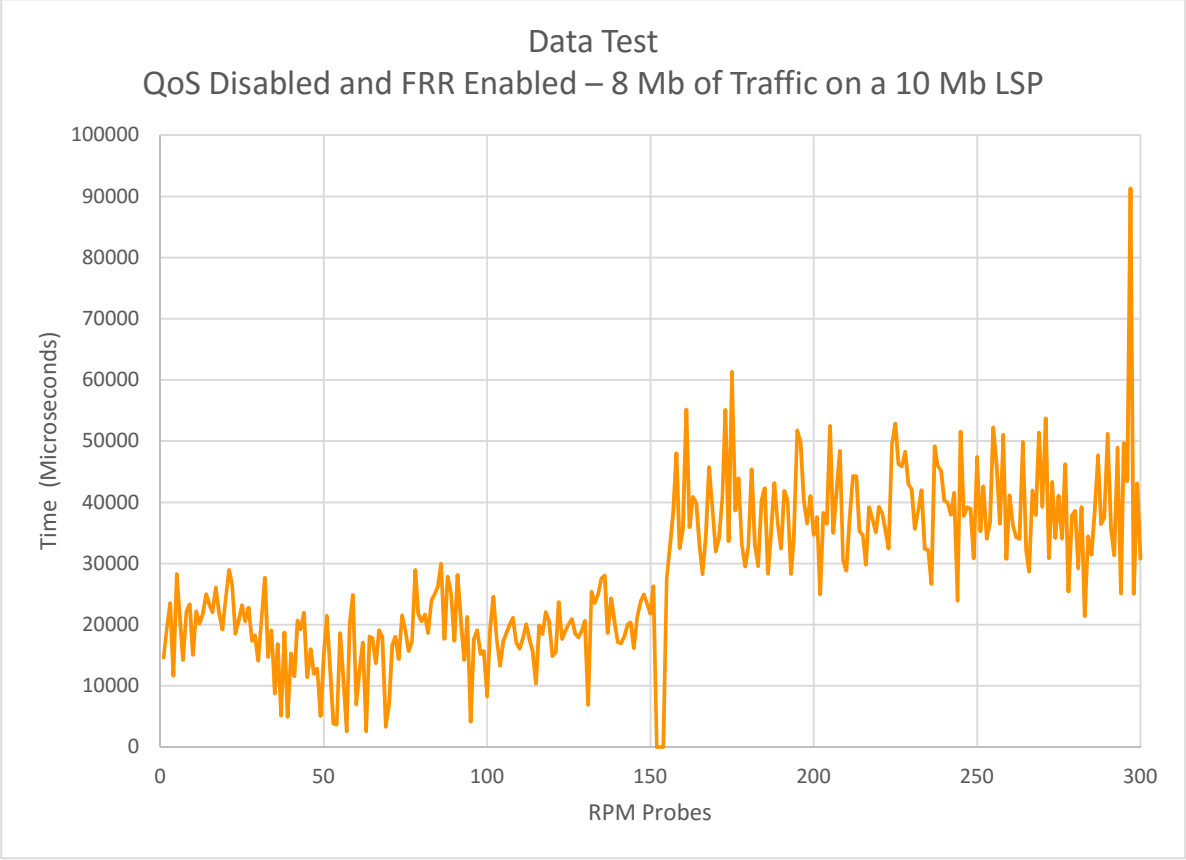

Figure 4.15: Performance of Data Traffic - QoS Disabled and FRR Enabled – 8 Mb of Traffic on a 10 Mb LSP

Since QoS has been disabled we see all traffic being classified as Best Effort. We see low end to end latency even though Quality of Service has been disabled although the packet sizes are larger than video and voice. An observation for this behavior could be the virtual routers are receiving traffic lower than their paths maximum bandwidth carrying capacity. When traffic fails onto the secondary link we see a higher round trip time value due to extra hops in the path to the egress router. We can see also a sudden spike in round trip time during the last few probes sent out.

When we compare the above scenario with Scenario 7, we can see that as long as the virtual routers are carrying traffic below the maximum bandwidth carrying capacity, QoS

does not really add a lot of improvement. A small improvement is however seen when traffic flows across increased hops. It is during period that we see the true benefits of Quality of Service.

Another observation that can be pointed out with respect to this test, is the reduction in packet losses during failover. In tests where Fast Reroute has been disabled, we can see that as traffic fails from the primary path onto the secondary path three to four RPM probes are lost. However, in the case of data traffic being sent at less than the paths maximum bandwidth carrying capacity we see only two packet losses when Fast Reroute is running. Based on the data application being used, TCP could resend the lost packets seen during failover.

4.6 Scenario 6: QoS Disabled and FRR Enabled – 12Mb of Traffic on a 10 Mb LSP

Scenario 6 compares three different types of traffic, namely voice, video and data traffic on a congested path i.e. 12 Mb of traffic on a 10 Mb label switched path. In this scenario, tests were carried out with Quality of Service being disabled and Fast Reroute being enabled.

4.6.1 Voice Traffic

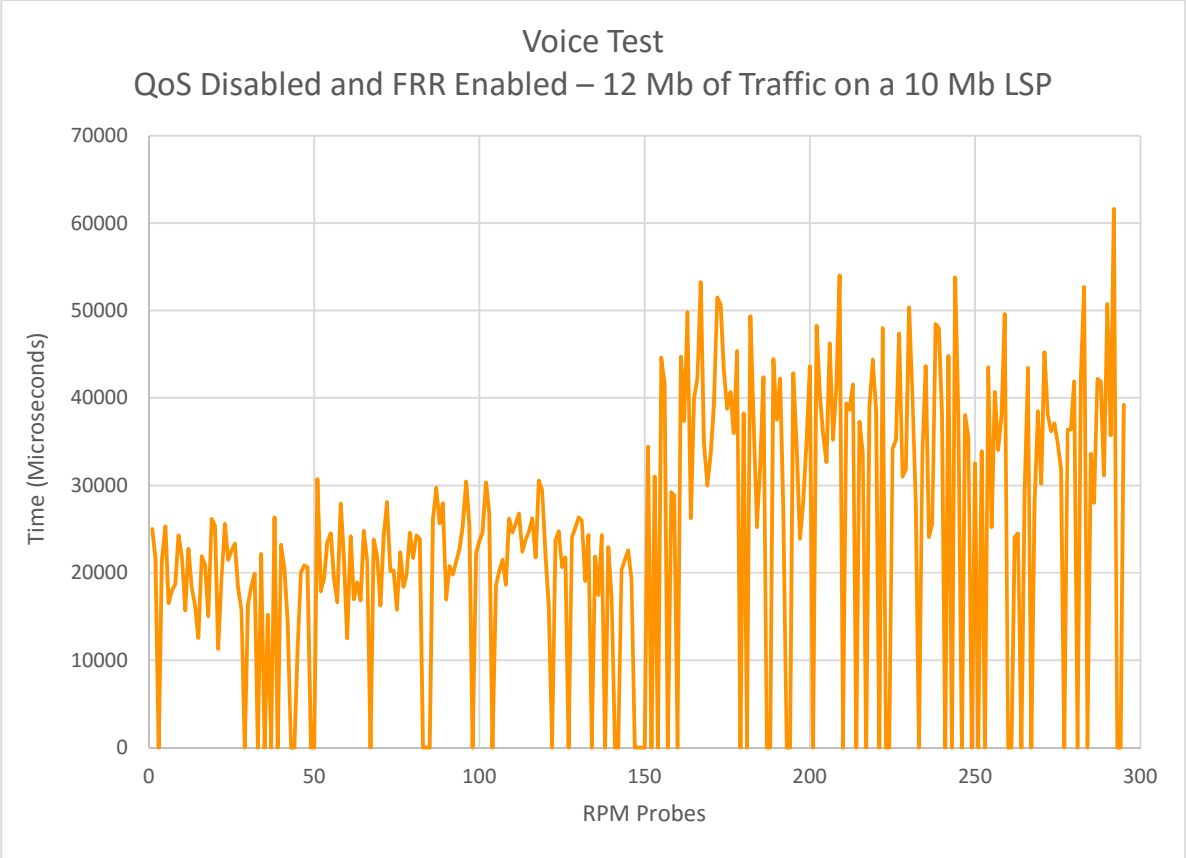

Figure 4.16: Performance of Voice Traffic - QoS Disabled and FRR Enabled – 12 Mb of Traffic on a 10 Mb LSP

In Scenario 6, QoS has been disabled while voice, video, data are all treated as best effort traffic. Traffic is flowing at more than the paths traffic carrying capacity. An observation from the above graph can be made where since voice traffic packet sizes are

relatively small we can see short RTT values. However, since voice traffic is contending with data and voice traffic for bandwidth a number of RTT probe timeouts occur, with an increase when sent on both the primary and secondary paths. When we disable QoS we see reduced RTT probe timeouts when compared against Scenario 8 where QoS has been enabled. The reason tends to be, QoS rate-limits the traffic every time it bursts over the allowed threshold.

Similar to the remaining scenarios where Fast Reroute has been enabled in a virtualized service provider network, its performance gain is not seen. Due to congestion on the link, we still see 4 RTT probe losses during failover. It would be interesting to see if the packet loss values remain consistent or decreases when using virtual routers such as vMX or Cisco Systems Advanced Service Routers. This has been mentioned in the future section of this thesis.

4.6.2 Video Traffic

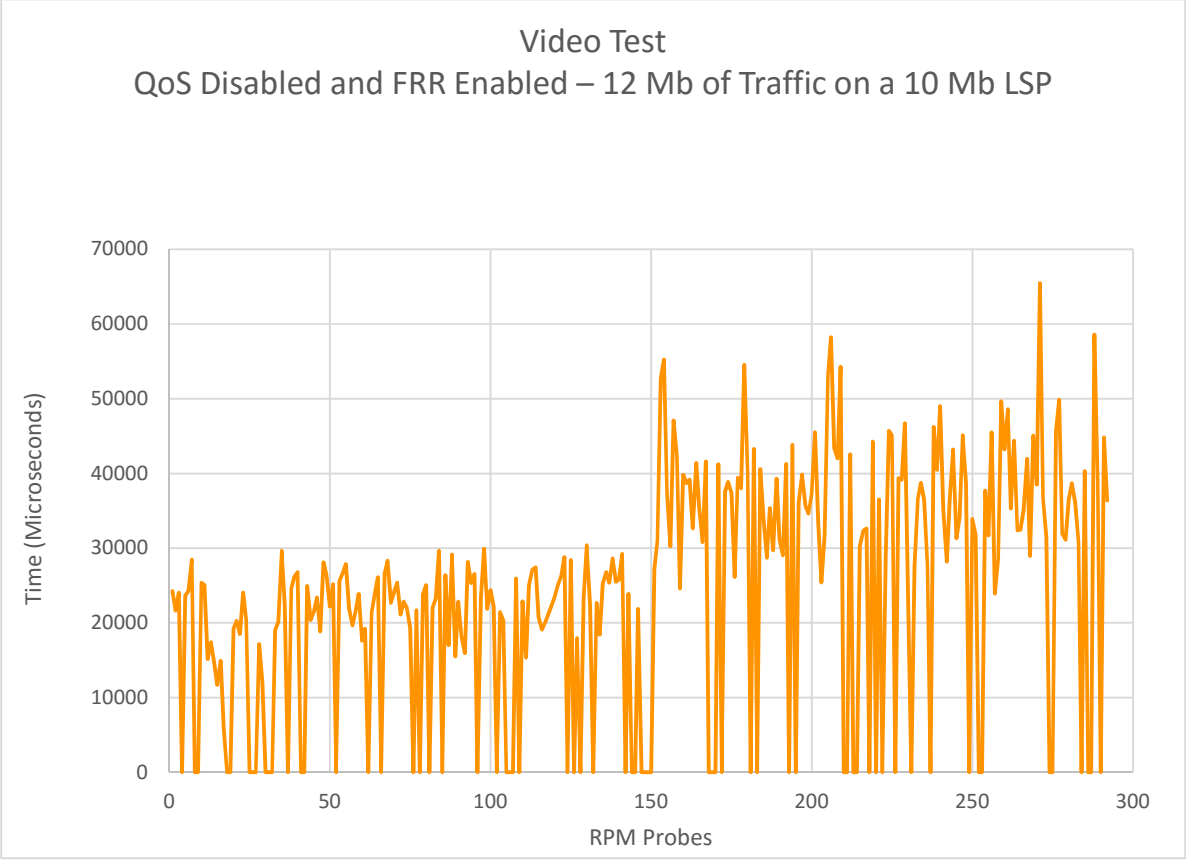

Figure 4.17: Performance of Video Traffic - QoS Disabled and FRR Enabled – 12 Mb of Traffic on a 10 Mb LSP

In Scenario 6, video, voice and data traffic are being treated as best effort since QoS has been disabled. Traffic is being sent at a rate higher than the paths maximum bandwidth carrying capacity. When we compare the same scenario with Scenario 5, we can see higher round-trip time probe losses. This tends to be due to path congestion.

Whilst comparing the same scenario with Scenario 8, an observation can be made that when QoS has been disabled the path can carry more video traffic while keeping the end to end latency almost similar to a QoS enabled scenario. However, this would be at the expense of affecting the performance of voice and data traffic. On the other we can see that when QoS

has been enabled, due to traffic crossing its maximum burst size the traffic starts getting hard policed, which in turn leads to higher Round Trip Time Probe failures but at that same time guarantees certain amount of transmission rate to other types of traffic.

Unfortunately, even during periods of traffic congestion we see that Fast Reroute fails to reduce packet loss during the failover from the primary to secondary path in a virtualized network yielding the same amount of RTT probe losses when fast reroute has been disabled.

4.6.3 Data Traffic

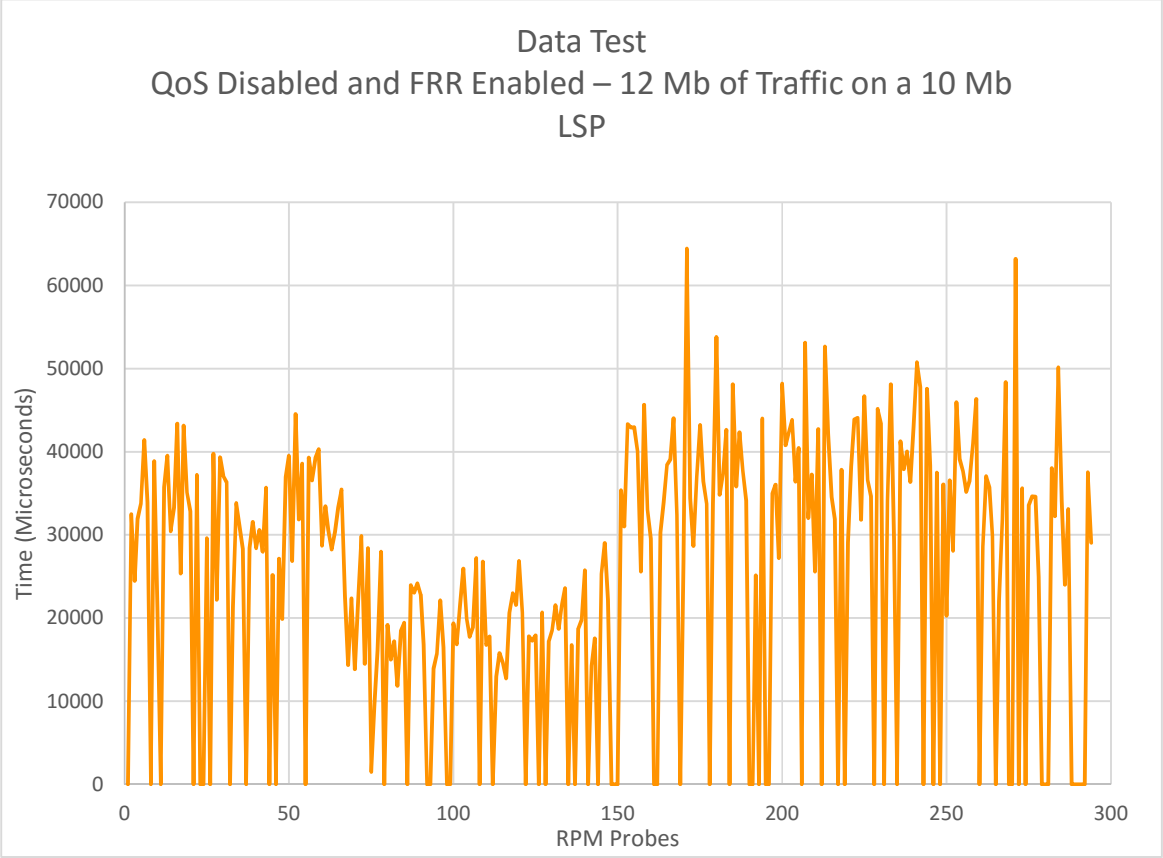

Figure 4.18: Performance of Data Traffic - QoS Disabled and FRR Enabled – 12 Mb of Traffic on a 10 Mb LSP

In Scenario 6, data traffic is being mixed with voice and video traffic as Quality of Service has been disabled. An observation from the above graph can be made wherein we see that when traffic exceeds the paths maximum bandwidth carrying capacity, virtual routers tend to maintain similar round trip times on both the primary and secondary paths. We also see a number of round trip time probe losses which occurs due to path congestion. Comparing the above scenario with scenarios where Quality of Service has been enabled we observe having QoS disabled tends to reduce RTT probe timeouts but increase the end to end

latency, where as having QoS enabled tends to drop higher amounts of traffic, i.e. traffic exceeding the policer while reducing the round-trip times.

We can also see that due to path congestion and excessive overhead on the router processing packets, the Fast Reroute feature cannot aid in reducing packet loss. Tests results during failover are identical to other scenarios where Fast Reroute has been disabled.

4.7 Scenario 7: QoS Enabled and FRR Enabled – 8Mb of Traffic on a 10 Mb LSP

Scenario 7 compares three different types of traffic, namely voice, video and data traffic on a non-congested path i.e. 8 Mb of traffic on a 10 Mb label switched path. In this scenario, tests were carried out with Quality of Service and Fast Reroute being enabled.

4.7.1 Voice Traffic

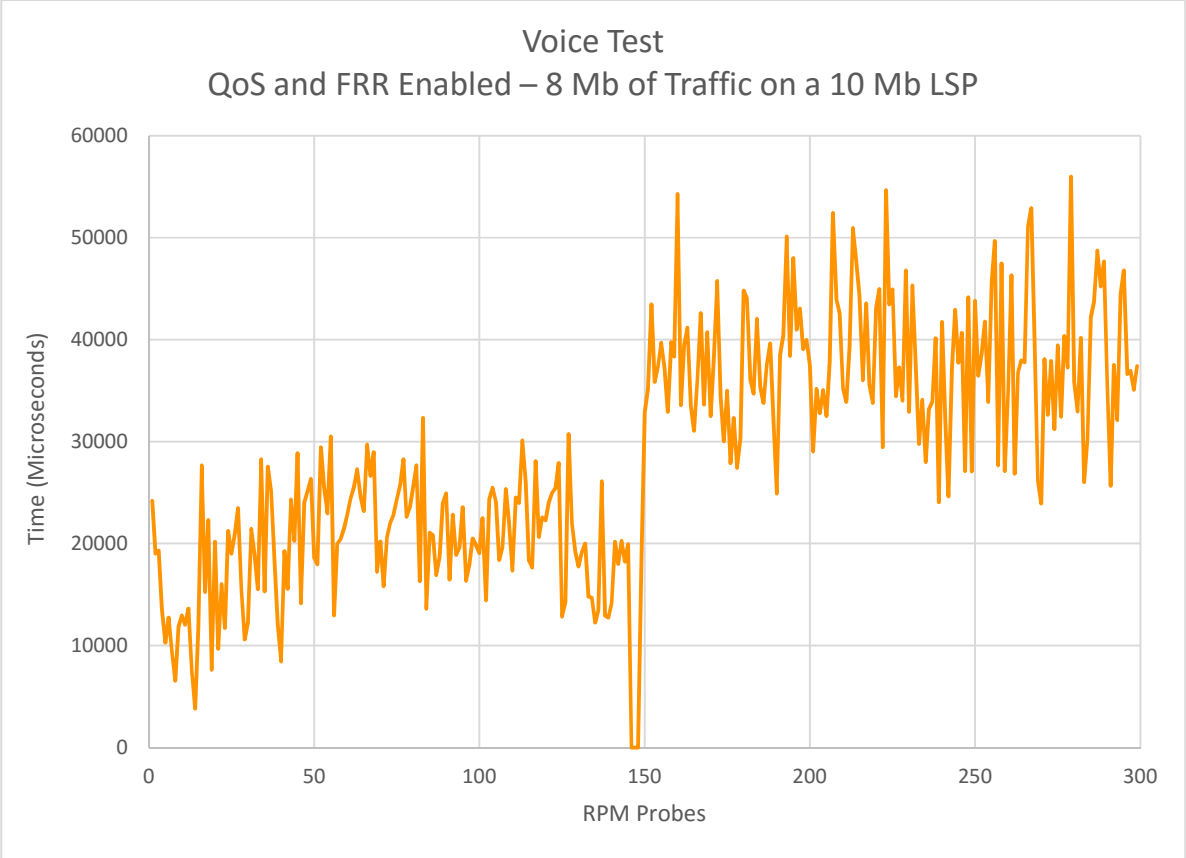

Figure 4.19: Performance of Voice Traffic - QoS and FRR Enabled – 8 Mb of Traffic on a 10 Mb LSP

Scenario 7 shows similar results when compared with Scenario 3. In both the scenario's QoS has been enabled and traffic is flowing at less than the paths maximum bandwidth carrying capacity. We can see from the above graph that RTT values tend to

remain below the 30000-microsecond benchmark, whilst on the secondary path, below 55000 microseconds.

The benefits of Quality of Service for voice traffic can be seen when comparing the grap above with Scenario 5's voice traffic test. The average Round Trip Times are reduced (lesser is better) in Scenario 7 where QoS has been enabled.

Unfortunately, when we compare the performance against Scenario 3, where QoS has been enabled but Fast Reroute has been disabled, we fail to see major performance improvements in terms of reducing packet losses during times of failover between the primary and secondary paths. In scenario 7, we still see three packet losses during failover.

4.7.2 Video Traffic

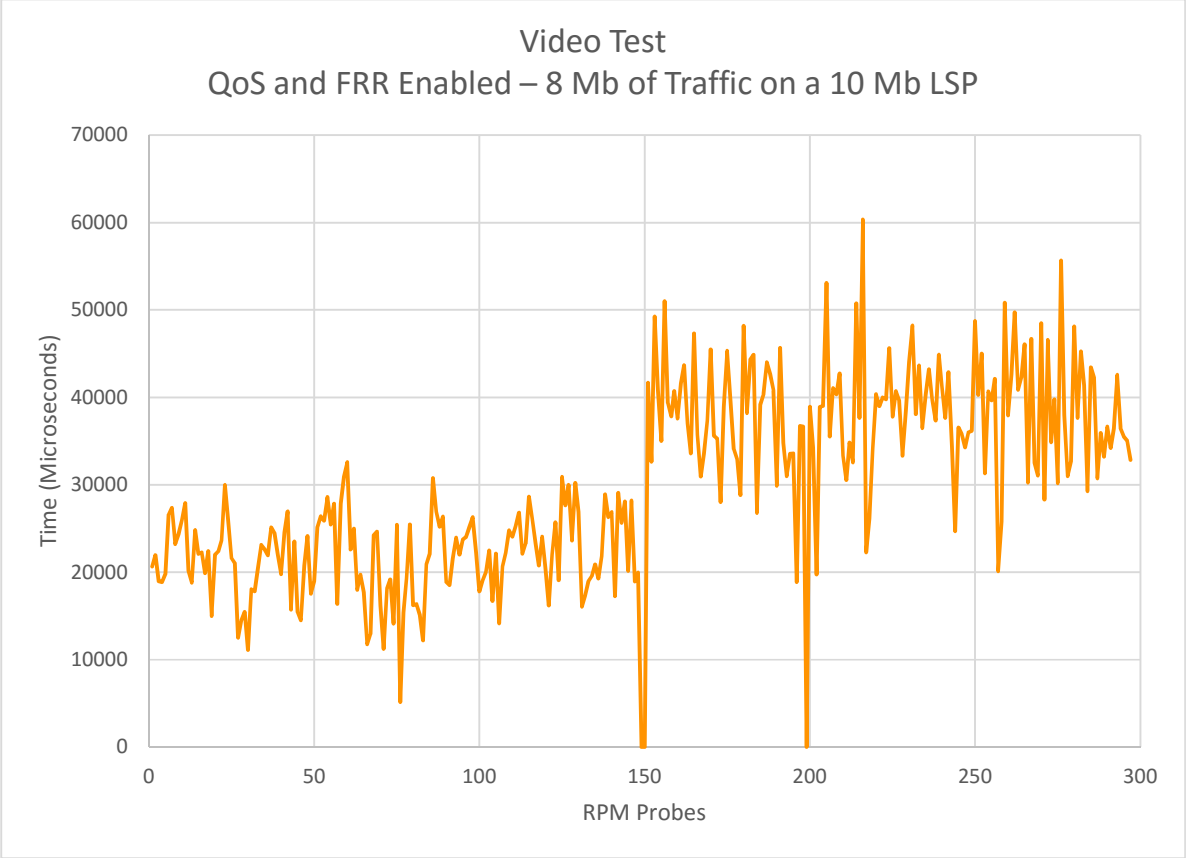

Figure 4.20: Performance of Video Traffic - QoS and FRR Enabled – 8 Mb of Traffic on a 10 Mb LSP

In Scenario 7, we see that QoS has been enabled with video traffic getting its own dedicated transmit rate. We see that when traffic flows on the primary path, the Round-Trip times tend to be consistent. We can see a small increase in the Round-Trip timers when traffic flows over the secondary path. This tends to be due to the increase in the number of next hops. When we compare Scenario 7 with Scenario 5 (QoS Disabled) we observe that performance tends to be similar. However, when congestion occurs (Scenario 6 and 8) we can see that scenario 7 fairs better due to Quality of Service being enabled and the average round trip times are lower as well as reduced round-trip time timeouts.

When traffic fails from the primary path onto the secondary path we can see only two round trip time timeouts. This shows us that the fast reroute feature has been able to detect this path failure and redirect traffic temporary over a detour path until traffic could finally fallback over the secondary path.

4.7.3 Data Traffic

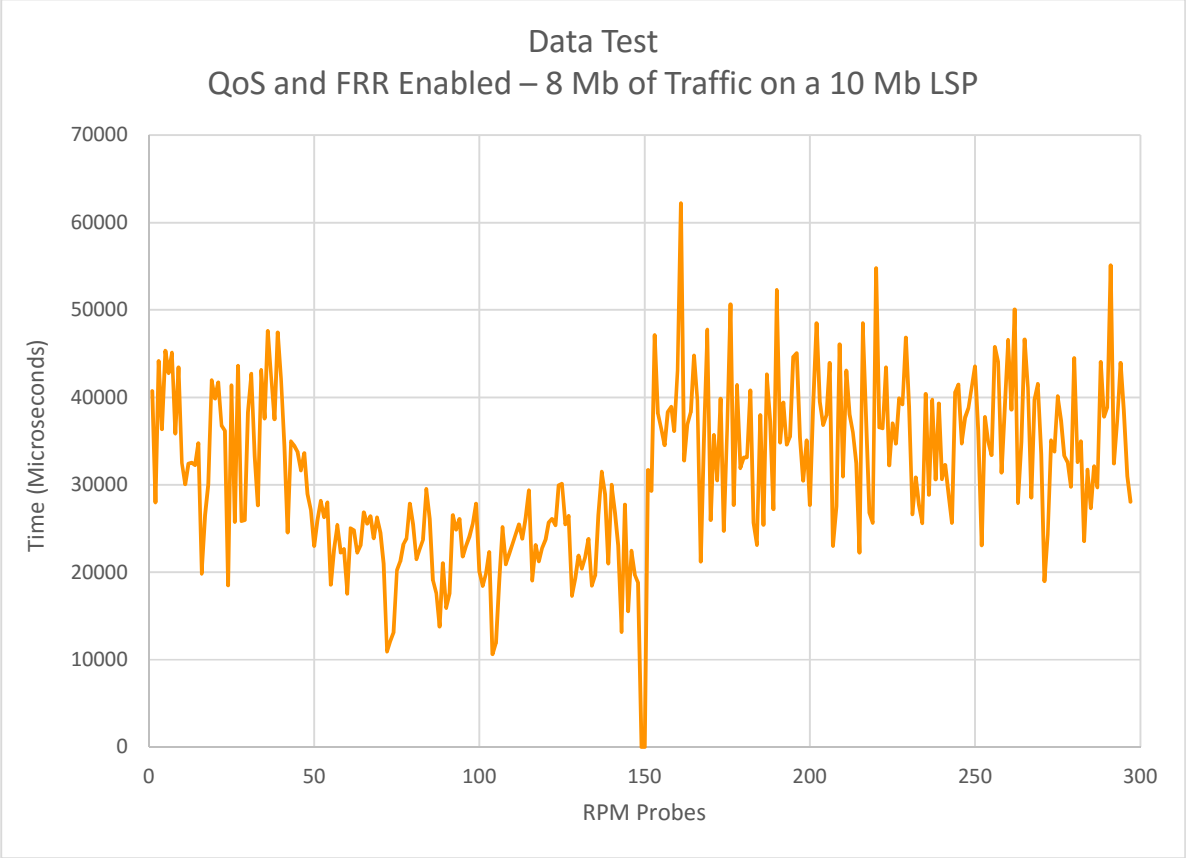

Figure 4.21: Performance of Data Traffic - QoS and FRR Enabled – 8 Mb of Traffic on a 10 Mb LSP

Since QoS, has been enabled, data traffic is being classified as Best Effort. Although the packet size of data traffic is larger than voice and video, we can see low round-trip time values. An observation for this behavior could be the virtual routers are receiving traffic lower than their paths maximum bandwidth carrying capacity. When traffic fails onto the secondary link we see a higher round-trip time value due to extra hops in the path to the egress router.

When we compare the above scenario with Scenario 5, we can see that as long as the virtual routers are carrying traffic below the maximum bandwidth carrying capacity QoS

does not reduce the average round-trip times to a great extent. However, a small improvement is seen when traffic flows across the secondary path.

What is surprising is to see the Fast Reroute feature aiding in minimizing packet loss during failover. Unlike scenarios where Fast Reroute has been disabled and 4 packet losses are seen we can observe only two packet losses in this scenario. Based on the data application being used, TCP could resend the lost packets seen during failover.

4.8 Scenario 8: QoS Enabled and FRR Enabled – 12Mb of Traffic on a 10 Mb LSP

Scenario 8 compares three different types of traffic, namely voice, video and data traffic on a congested path i.e. 12 Mb of traffic on a 10 Mb label switched path. In this scenario, tests were carried out with Quality of Service and Fast Reroute being enabled.

4.8.1 Voice Traffic

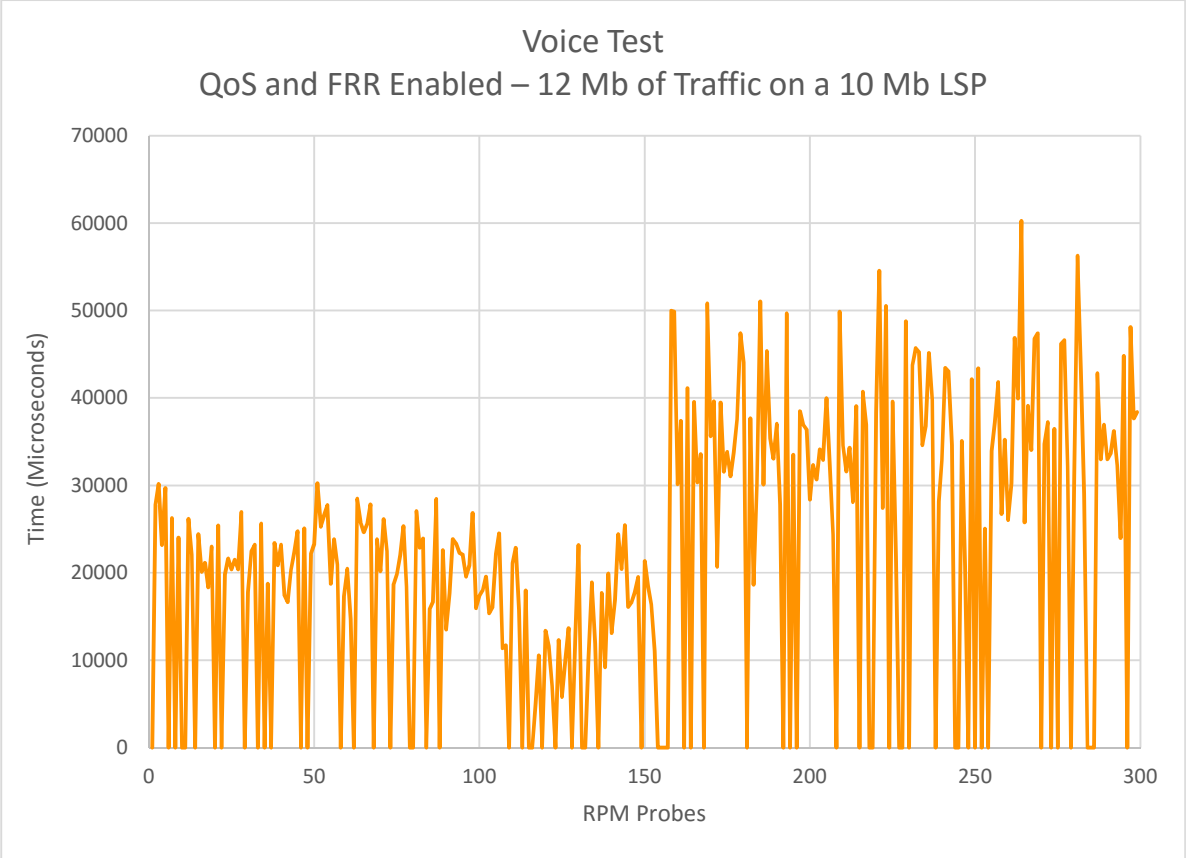

Figure 4.22: Performance of Voice Traffic - QoS and FRR Enabled – 12 Mb of Traffic on a 10 Mb LSP

In Scenario 8, QoS has been enabled and voice traffic is being sent along with data and video traffic at an aggregate bandwidth of 12 Mb. We can see the trend in the graph showing us RTT probe losses occur every time voice exceeds its allocated bandwidth limit or

when the buffer is full. The results of this test are similar to Scenario 4, wherein we can see similar trends in the graph.

Again, in times of congestion, we see an additional packet loss during failover from the primary path onto the secondary path. An assumption for fast reroute not working effectively by reducing packet losses during failover can be made which shows us that since the routers are already over-worked processing traffic on a congested label switched path, it fails to minimize packet drops in a virtualized network. It would be interesting to see the performance of voice traffic with different scheduling parameters such as increased buffer sizes configured. This has been mentioned in future direction section of this thesis.

4.8.2 Video Traffic

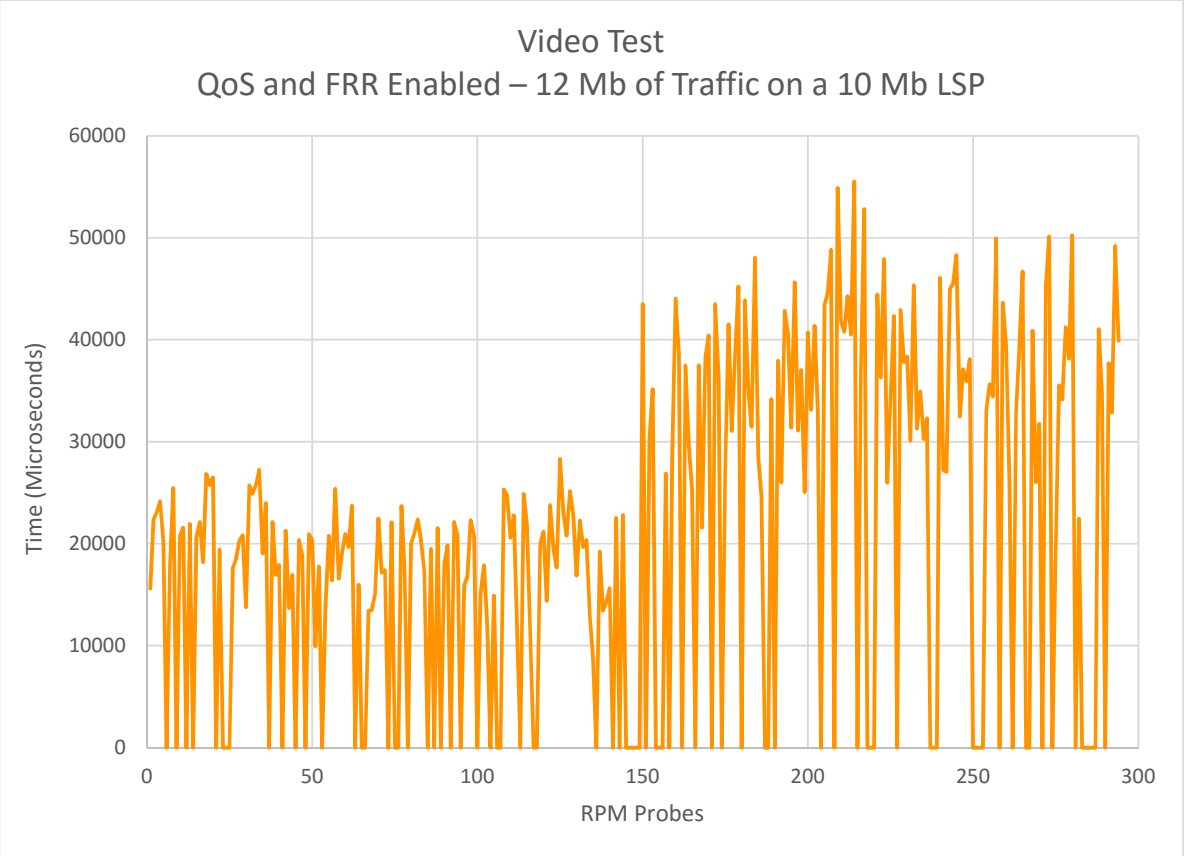

Figure 4.23: Performance of Data Traffic - QoS and FRR Enabled – 12 Mb of Traffic on a 10 Mb LSP

In the above graph, we can see the performance of Video traffic when traffic is being pushed more than the paths bandwidth carrying capacity. An observation from the above graph can be made that when video traffic exceeds it maximum permitted capacity, traffic is being dropped due to the actions configured in the policer. This can be reflected in graph showing us a number of round trip time timeouts.

The above graph also shows us that approximately six round trip time probes are lost, before traffic gets to flow on the secondary path. An assumption can be made wherein the virtual routers processes are already overworked, it fails to minimize packet loss using Fast Reroute.

4.8.3 Data Traffic

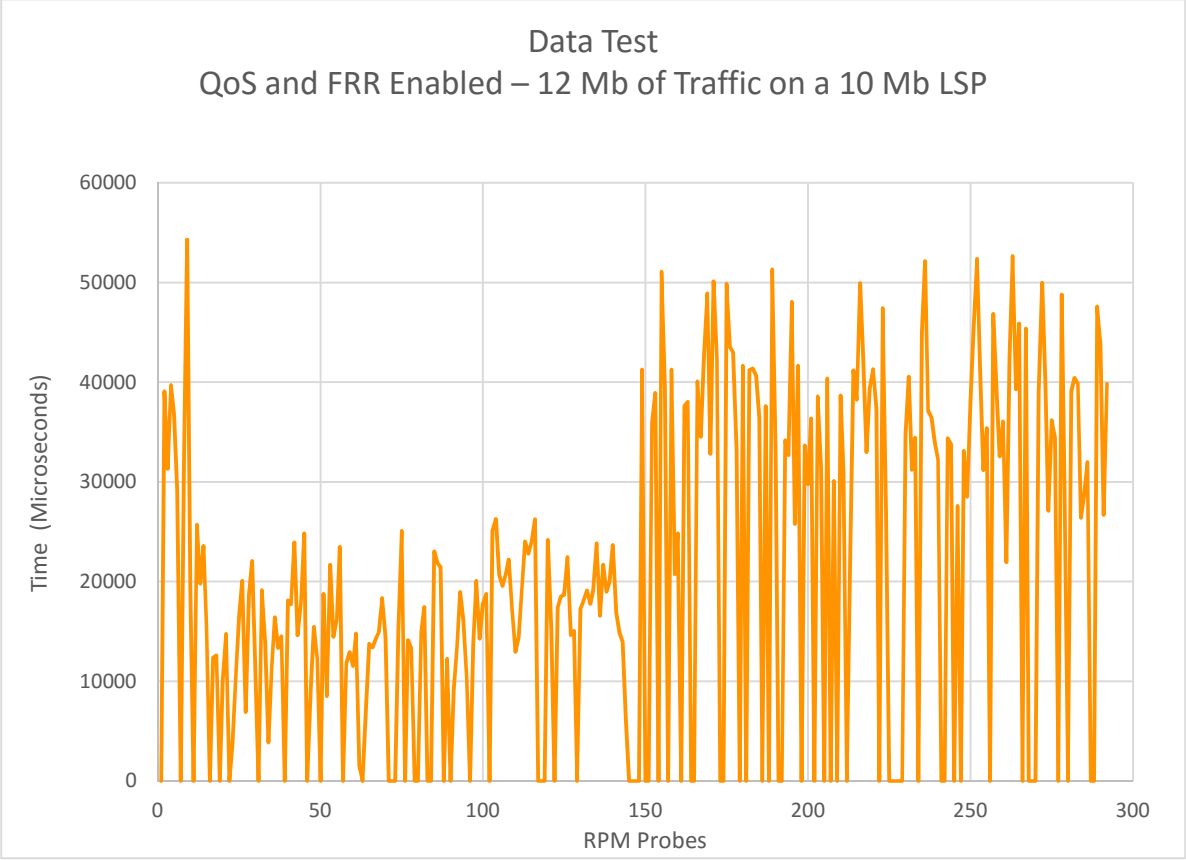

Figure 4.24: Performance of Data Traffic - QoS and FRR Enabled – 12 Mb of Traffic on a 10 Mb LSP

In Scenario 8, data traffic is being classified as Best Effort since Quality of Service has been enabled. We can see from the above graph that when data traffic crosses the committed burst size, violating packets are policed and dropped. When we compare this scenario with scenario 6, we see Quality of Service reducing the average RTT for traffic flowing on the primary path. However, this is at the expense of dropping more data traffic as compared to Scenario 6, where Quality of Service has been disabled. Although this leads to lesser traffic being sent between the source and destination, it tends to improve the end to end delay taken by for in-profile traffic. We also see a number of round trip time probe losses

which occurs due to path congestion. Comparing the above scenario with scenarios where Quality of Service has been disabled we see having QoS disabled tends to reduce RTT probe timeouts but increase the end to end latency, whereas having QoS enabled tends to drop higher amounts of traffic, i.e. traffic exceeding the policer while reducing the average round trip times.

Another observation which can be made is data traffic tends to benefit from Fast Reroute. As seen in tests above, when data traffic is being used, Fast Reroute tends to reduce the number of round trip time probe losses. As compared to the other types of traffic which drop 6 packets during failover with congestion present, data traffic only losses 4 packets during congestion with Fast Reroute being enabled.

5 Conclusion

In this thesis, I have evaluated the performance of data and real-time (voice and video) traffic in a virtualized MPLS core. The performance metric which helped me evaluate the same has been Round-Trip Time. Eight different scenarios consisting of twenty-four different tests have been simulated to gain a better understanding of the performance of MPLS in a virtualized network. A comparison has been made of how data and real-time traffic is affected when sent across congested and un-congested paths with/without Quality of Service. Further tests have also been performed to understand if MPLS resiliency features such as Fast Reroute can aid in reducing packet loss during failovers between primary and secondary paths in a virtualized MPLS network.

In Section 1.3 of this thesis, four research questions were made to examine the performance of MPLS traffic in a virtualized network. The answers to those questions are mentioned below.

[1] How does data and real-time traffic perform when Quality of Service is disabled across congested and uncongested paths?

After analyzing the simulation results for quality of service disabled scenarios we can observe that, for uncongested paths i.e. when traffic being sent is lesser than the paths maximum bandwidth carrying capacity, the graph trend seems to be consistent for voice, video and data traffic. Traffic routed across primary links tend to have average RTT values around 20000 microseconds. Video and data traffic seem to perform better than voice traffic over primary paths. This observation is made by comparing the lowest average Round-Trip Time. For traffic flowing across the secondary path, we see consistent results for all three traffic types. In certain scenarios, we can see one or two packet losses for video traffic. When comparing uncongested link scenarios vs congested scenarios, the latter's graph trend show us higher RTT values which remain constant for traffic flowing across both primary and

secondary paths. Traffic flowing across congested paths having QoS disabled would severely affect an applications performance, typically real-time applications which do not re-send dropped packets.

[2] How does data and real-world traffic perform when Quality of Service is enabled across congested and uncongested paths?

After analyzing the simulation results for Quality of Service enabled scenarios we can observe that for uncongested paths, we notice RTT values are the lowest for voice traffic throughout the primary path. After voice, the next best performing traffic is data. Video traffic compared to voice and data has the highest RTT values on the primary path. All three traffic types perform relatively similar when being sent on the uncongested secondary path. As long as traffic is below the paths makes bandwidth carrying capacity we do not see any packet loss using QoS except during the failover between primary and secondary paths.

For traffic flowing on congested paths, we can see how QoS discards traffic exceeding its committed burst size. A higher number of packet drops are seen when QoS has been enabled on congested links. Again, voice traffic seems to perform better than video and data traffic having an overall lower RTT value. When comparing uncongested link scenarios vs congested scenarios, the latter's graph trend shows us higher RTT values which remain constant for traffic flowing across both primary and secondary paths.

[3] Does Quality of Service improve the overall performance on congested and uncongested paths in a virtualized MPLS network?

When traffic is passed across the primary uncongested label switched path we observe the performance achieved by enabling QoS is the same as having QoS disabled for real-time traffic. However, data traffic seems to have lower RTT values when QoS is disabled! Again, for traffic passing on the secondary (increased hop) uncongested label switched path we

observe a tie in performance for real world traffic between QoS enabled and disabled scenarios.

A trade-off exists for introducing QoS on congested primary and secondary label switched paths. What can be observed thru the graphs is having Quality of Service enabled drops more packets however gives us the advantage of lower Round Trip Time for in-profile traffic. On the hand, having Quality of Service disabled, permits more traffic but leads to contention between the three traffic classes leading to higher Round-Trip Times. The true benefit of QoS is seen in traffic congestion scenarios.

[4] Does Fast Reroute add significant resiliency in a virtualized MPLS network?

Using the testbed specification in Section 3.1.1 of this thesis, it is observed that Fast Reroute does not add a significant benefit to aid in the reduction of packet loss during failover scenarios between primary and secondary paths. However, in certain scenarios fast reroute seems to reduce packet loss specifically for data traffic. I believe virtualized carrier-grade routers such as Juniper Networks vMX or Cisco Systems Advanced Service Routers could perform this function better. Alternatively, increasing system resources for virtual routers may increase the performance of Fast Reroute in a virtualized core. However, I believe further research is required to analyze the performance of Fast Reroute in a virtualized MPLS network. Further information about the same can be found within the Future Direction section of this thesis.

Network Function Virtualization is an emerging technology and has application scope within several areas of the networking industry. However, more research needs to be conducted to understand the performance of applications and traffic in virtualized networks. I feel a tighter integration between hardware and software is necessary in order for this technology to be successful.

6 Future Direction

In the future, I would like to take my research a step further to gain a better understanding of Network Function Virtualization and the performance of real-time traffic in a virtualized MPLS environment. Some aspects I would be interested are:

- Hypervisor: The chosen hypervisor for this thesis is VMWare Fusion. I would like to see the performance and resiliency feature when alternate hypervisors such as QEMU/KVM or VMWare ESXi are used.
- Different Virtual Routers: At the time of writing this thesis, an evaluation version of Juniper Networks vMX was not yet available. This carrier-grade virtual router is specifically designed for use in Service Provider networks. I would like to see the performance of real-time applications in a virtualized core when the same tests are conducted on Juniper Networks vMX, Cisco Systems Advanced Services Router etc.
- Changed QoS Configuration and Real-Time Codecs: I would also like to experiment and analyze the performance of real-time applications with different QoS settings. In future versions of Virtual Routers, I would also like to see how real-time applications perform with more granular QoS settings such as three-level policers, increased buffer sizes and also the possibility of H-QoS.
- Distributed server setup with additional customers: I would also like to see if the performance of virtualized network environments can improve when resources are allocated across multiple servers. Additionally, I would like a scaled-up version of the topology with additional customers connecting to the PE's. I believe this would not only add overhead to the PE's but also allow us to see a virtual routers performance is affected within the core.
- Use of commercial-grade hardware traffic generators: In this thesis, I used Colasoft Packet Player to replay Wireshark captured packets. This application runs as a

software on the same server running the virtual routers. I would like to use commercial-grade traffic generators such as iXIA Traffic Generator which are hardware based and can push increased traffic flows/streams. iXIA's Traffic Generator would also give me additional performance parameters to track.

- Hardware and Software Improvements: NFV is still an emerging technology. If this technology needs to be successful, a tighter integration between hardware and software is required. Carrier-grade physical routers have dedicated line cards with hardware modules to achieve a certain level of service. I believe more research is needed on bringing these features onto servers to increase the performance of virtual routers. The operating system must also be optimized to integrate better with the hardware. vSRX makes use of a single image to run control plane and forwarding plane operations. I would like to see if the performance of these virtualized networks improves when the control plane and forwarding plane are hosted as separate virtual machines.

7 Bibliography

[1] Girish, M.K.; Bei Zhou; Jian Qiang Hu, "Formulation of the traffic engineering problems in MPLS based IP networks," in Computers and Communications, 2000. Proceedings. ISCC 2000. Fifth IEEE Symposium on, vol., no., pp.214-219, 2000

doi: 10.1109/ISCC.2000.860641

Available:

http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=860641&isnumber=18650

[2] AT&T. (2015). Why you still need an MPLS VPN (1st edition) [Online]. Available:

http://www.business.att.com/content/whitepaper/mpls-vpn.pdf

[3] S. H.Zwayen and M. B. Ibrahim, 'Evaluating the Performance of MPLS and Frame-Relay using OPNET Modeler', International Journal of Computer Applications, vol. 108, no. 12, pp. 32-34, 2014.

Available: http://research.ijcaonline.org/volume108/number12/pxc3900306.pdf

[4] D. Parmar and D. Patalia, Analysis Of QoS Enabled MPLS VPN VOIP NETWORK WITH RIPV2 ROUTING PROTOCOL, 1st ed. Gujarat: ISSN: 0975 – 6760 | Nov 12 To Oct 13 | Volume – 02, Issue – 02| Journal Of Information, Knowledge And Research In Computer Engineering, 2013, Pp. 404-407.

Available: http://www.ejournal.aessangli.in/ASEEJournals/CE83.pdf

[5] Jeevan Kharel and Deepak Adhikhari, “Performance Evaluation of Voice Traffic over MPLS Network with TE and QoS Implementation”, M.S. thesis, Dept. Electrical Engineering, Blekinge Institute of Technology, Karlskrona, Sweden, 2011. Available:http://bth.diva-portal.org/smash/get/diva2:832105/FULLTEXT01.pdf

[6] Muhammad Naeem Aslam and Yassar Aziz, “Traffic Engineering with Multi-Protocol Lab Switching - Performance Comparision with IP Networks”, M.S. thesis, Dept. Computer Science, Blekinge Institute of Technology, Ronneby, Sweden, August 2008. Available:http://bth.diva-portal.org/smash/get/diva2:833436/FULLTEXT01.pdf

[7] Yihan Li, Shivendra Panwar and CJ. Liu, “Performance Analysis of MPLS TE Queues for QoS Routing,”

[8] O. Gure, B. K. Boyaci, and N. O. Unverdi, “Analysis of the Service Quality on MPLS Networks,” In Circuits and Systems for Communications (ECCSC), 5th European Conference, pp. 43-46, 2010.

[9] H. Hodzic and S. Zoric, “Traffic Engineering with Constraint Based Routing in MPLS Networks,” In ELMAR, 50th International Symposium, IEEE, Vol. 1, pp. 269-272, 2008.

[10] S. Sharafali, M. Al-Quzwini and R. Fyath, 'Performance Evaluation of MPLS TE Signal Protocols for Voice Applications with QoS Implementation', International Journal of Networks and Communications, vol. 5, no. 1, pp. 1-9, 2015.

[11] Nasser Hadi Saleh Almofary, “Study of TV IP and VoIP Performance on IP, MPLS and ATM Networks” M.S. thesis, Electronics and Communication Department, Mansoura University, 2012.

[12] Sathappan Kathiresan, “Performance Analysis of MPLS Over IP Networks using Cisco IP SLAs”, M.S. thesis, School of Engineering Science, Simon Fraser University, Spring 2015.

[13] Bin Ali, Z.; Samad, M.; Hashim, H., "Performance comparison of video multicasting over Asynchronous Transfer Mode (ATM) & Multiprotocol Label Switching (MPLS) networks," in System Engineering and Technology (ICSET), 2011 IEEE International Conference. pp.177-182, 27-28 June 2011
doi: 10.1109/ICSEngT.2011.5993445
Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5993445&isnumber=5993403

[14] M. Aziz and M. Islam, "Performance Evaluation of Real–Time Applications over DiffServ/MPLS in IPv4/IPv6 Networks", Masters, Blekinge Institute of Technology, 2010. Available:
http://www.bth.se/com/mscee.nsf/attachments/Binder1_pdf/$file/Binder1.pdf

[15] Devara, Kanakeswari and Chokkapu Narayanarao. "Traffic Protection against Link and Node Failures Using Fast Re-Route for MPLS Networks." IJRCCT 3.8 (2014): 889-895. Available: http://www.ijrcct.org/index.php/ojs/article/viewFile/808/pdf

[16] Chandana B and Piruthiviraj P. "An Efficient QOS Routing Algorithm for Protection of Data Flow in MPLS Network"
International Journal of Innovative Research in Computer and Communication Engineering. vol. 2, no. 5, pp. 4328-4334, 2015
Available: http://www.ijircce.com/upload/2014/may/24_AnEfficient.pdf

[17] "RFC 791 - Internet Protocol", Tools.ietf.org, 1981. [Online]. Available: https://tools.ietf.org/html/rfc791.

[18] A. Blank, TCP/IP JumpStartTM, 1st ed. Hoboken: John Wiley & Sons, 2006.

[19] C. Kozierok, The TCP/IP-Guide, 1st ed. No Starch Press, p. 607.

[20]I. Minei and J. Lucek, *MPLS-Enabled Applications: Emerging Developments and New Technologies, 2nd*, 1st ed. John Wiley & Sons, 2008.

[21]T. Support, "MPLS FAQ For Beginners", *Cisco*, 2016. [Online]. Available: http://www.cisco.com/c/en/us/support/docs/multiprotocol-label-switching-mpls/mpls/4649-mpls-faq-4649.html#anc1. [Accessed: 18- Apr- 2017].

[22]"How packets are forwarded through an MPLS domain", *Brocade.com*. [Online]. Available: http://www.brocade.com/content/html/en/configuration-guide/netiron-05900-

mplsguide/GUID-9BE9AAA9-5CC5-4A7E-B125-23CF171C1DCC.html. [Accessed: 18-Apr- 2017].

[23]"Multiprotocol Label Switching (MPLS) Introduction :: Chapter 1. Multiprotocol Label Switching (MPLS) Architecture Overview :: Part I: MPLS Technology and Configuration :: MPLS and VPN Architectures :: Networking :: eTutorials.org", *Etutorials.org*, 2017. [Online]. Available: http://etutorials.org/Networking/MPLS+VPN+Architectures/Part+I+MPLS+Technology+and+Configuration/Chapter+1.+Multiprotocol+Label+Switching+MPLS+Architecture+Overview/Multiprotocol+Label+Switching+MPLS+Introduction/. [Accessed: 19- Apr- 2017].

[24]R. Aggarwal, K. Kompella, T. Nadeau and G. Swallow, "draft-ietf-bfd-mpls-02 - Bidirectional Forwarding Detection (BFD) for MPLS Label Switched Paths (LSPs)", *Tools.ietf.org*, 2005. [Online]. Available: https://tools.ietf.org/html/draft-ietf-bfd-mpls-02. [Accessed: 09- Apr- 2017].

[25]R. Braden, L. Zhang, S. Berson and S. Herzog, "RFC 2205 - Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", *Tools.ietf.org*, 1997. [Online]. Available: https://tools.ietf.org/html/rfc2205. [Accessed: 19- Apr- 2017].

[26] "Understanding The RSVP Signaling Protocol - Technical Documentation - Support - Juniper Networks". Juniper.net. https://www.juniper.net/techpubs/en_US/junos/topics/concept/mpls-security-rsvp-signaling-protocol-understanding.html. [Accessed: 21- Apr- 2017].

[27] "Link Protection - Technical Documentation - Support - Juniper Networks", Juniper.net. [Online]. Available: https://www.juniper.net/documentation/en_US/junos/topics/concept/mpls-link-protection-overview.html. [Accessed: 21- Apr- 2017].

[28] "Fast Reroute Overview - Technical Documentation - Support - Juniper Networks", Juniper.net. [Online]. Available: https://www.juniper.net/documentation/en_US/junos/topics/concept/mpls-fast-reroute-overview.html. [Accessed: 21- Apr- 2017].

[29] K. Jannu and R. Deekonda, "OPNET simulation of voice over MPLS with Considering Traffic Engineering.", Masters, Blekinge Tekniska Högskola, 2010.

[30] B. Anjum and H. Perros, Bandwidth Allocation for Video under Quality of Service Constraints, 1st ed. John Wiley & Sons, 2015.

[31] K. Nicholas, S. Blake, F. Baker and D. Black, "RFC 2474 - Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", Tools.ietf.org, 1998. [Online]. Available: https://tools.ietf.org/html/rfc2474. [Accessed: 24- Apr- 2017].

[32] J. Moy, "OSPF Version 2Cite a Website - Cite This For Me", Ietf.org, 1998. [Online]. Available: https://www.ietf.org/rfc/rfc2328.txt. [Accessed: 24- Apr- 2017].

[33] "OSPF Support for Traffic Engineering - Technical Documentation - Support - Juniper Networks", Juniper.net, 2016. [Online]. Available:

https://www.juniper.net/documentation/en_US/junos/topics/concept/ospf-traffic-engineering-support-overview.html. [Accessed: 23- Apr- 2017].

[34] "Loop-Free Alternate Routes for OSPF Overview - Technical Documentation - Support - Juniper Networks", Juniper.net, 2013. [Online]. Available: https://www.juniper.net/documentation/en_US/junos12.1x46/topics/concept/ospf-loop-free-alternate-routes-overview.html. [Accessed: 23- May- 2017].

[35] D. Katz and D. Ward, "RFC 5880 - Bidirectional Forwarding Detection (BFD)", Tools.ietf.org, 2010. [Online]. Available: https://tools.ietf.org/html/rfc5880. [Accessed: 24- Apr- 2017].

[36] Y. Rekhter, T. Li and S. Hares, "RFC 4271 - A Border Gateway Protocol 4 (BGP-4)", Tools.ietf.org, 2006. [Online]. Available: https://tools.ietf.org/html/rfc4271. [Accessed: 24- Apr- 2017].

[37] T. Bates, R. Chandra, D. Katz and Y. Rekhter, "RFC 4760 - Multiprotocol Extensions for BGP-4", Tools.ietf.org, 2007. [Online]. Available: https://tools.ietf.org/html/rfc4760. [Accessed: 24- Apr- 2017].

[38] Junos MPLS and VPNs, 12th ed. Juniper Worlwide Education Services, 2013.

[39] "VMware Fusion 8 Documentation Center", Pubs.vmware.com. [Online]. Available: http://pubs.vmware.com/fusion-8/index.jsp#com.vmware.fusion.using.doc/GUID-F4C352A0-16AA-476F-BE34-1FF2C5399DA4.html. [Accessed: 25- Apr- 2017].

[40] "VMware Fusion 8 Documentation Center", Pubs.vmware.com, 2017. [Online]. Available: http://pubs.vmware.com/fusion-8/index.jsp?topic=%2Fcom.vmware.fusion.using.doc%2FGUID-2B2EE44A-C3A0-42BF-8BF2-BC2C84CFD605.html. [Accessed: 25- Apr- 2017].

[41] "Junos Network Operating System - Juniper Networks", Juniper.net. [Online]. Available: https://www.juniper.net/us/en/products-services/nos/junos/. [Accessed: 25- Apr- 2017].

[42]"RPM Overview - Technical Documentation - Support - Juniper Networks", Juniper.net, 2015. [Online]. Available: https://www.juniper.net/documentation/en_US/junos12.1x46/topics/concept/security-rpm-overview.html. [Accessed: 25- Apr- 2017].

[43] "vSRX Release 12.1X47 (formerly Firefly Perimeter) - Technical Documentation - Support - Juniper Networks", Juniper.net. [Online]. Available: http://www.juniper.net/techpubs/en_US/firefly12.1x47/information-products/pathway-pages/security-virtual-perimeter-software-version-index.html. [Accessed: 25- Apr- 2017].

[44] "Firefly Perimeter Getting Started Guide for VMware - Technical Documentation - Support - Juniper Networks", Juniper.net, 2015. [Online]. Available: http://www.juniper.net/techpubs/en_US/firefly12.1x47/information-products/pathway-pages/security-virtual-perimeter-vmware-gs-guide-pwp.html. [Accessed: 25- Apr- 2017].

[45] "Firefly Suite Getting Started Guide - Technical Documentation - Support - Juniper Networks", Juniper.net, 2015. [Online]. Available: http://www.juniper.net/techpubs/en_US/release-independent/firefly/information-products/pathway-pages/security-firefly-suite-gsg.html. [Accessed: 26- Apr- 2017].

[46] "Packet Player for Network Engineer - Colasoft", *Colasoft.com*. [Online]. Available: http://www.colasoft.com/packet_player/. [Accessed: 26- Apr- 2017].

[47]"Tcpreplay", Tcpreplay.synfin.net, 2014. [Online]. Available: http://tcpreplay.synfin.net. [Accessed: 26- Apr- 2017].

[48]"tcprewrite – Tcpreplay", Tcpreplay.synfin.net, 2014. [Online]. Available: http://tcpreplay.synfin.net/wiki/tcprewrite. [Accessed: 26- Apr- 2017].

[49] "Wireshark · Go Deep.", Wireshark.org. [Online]. Available: https://www.wireshark.org. [Accessed: 26- Apr- 2017].

[50] "FileZilla - The free FTP solution", Filezilla-project.org. [Online]. Available: https://filezilla-project.org. [Accessed: 28- Apr- 2017].

[51] "FileZilla - Client Features", Filezilla-project.org. [Online]. Available: https://filezilla-project.org/client_features.php. [Accessed: 28- Apr- 2017].

[52] "Visio Pro for Office 365 |Office | Microsoft", Products.office.com. [Online]. Available: https://products.office.com/en-us/Visio/microsoft-visio-pro-for-office-365?wt.srch=1&wt.mc_id=AID522514_SEM_4xYGC7RC. [Accessed: 28- Apr- 2017].

[53] "Microsoft Excel 2016 Spreadsheet Software, Excel Free Trial", Products.office.com, 2017. [Online]. Available: https://products.office.com/en-us/excel. [Accessed: 28- Apr- 2017].

[54] "Features - iTerm2 - macOS Terminal Replacement", Iterm2.com. [Online]. Available: https://www.iterm2.com/features.html. [Accessed: 28- Apr- 2017].

[55]"TextEdit", En.wikipedia.org. [Online]. Available: https://en.wikipedia.org/wiki/TextEdit. [Accessed: 28- Apr- 2017].

[56] "OS X El Capitan", En.wikipedia.org. [Online]. Available: https://en.wikipedia.org/wiki/OS_X_El_Capitan. [Accessed: 29- Apr- 2017].

[57] "About the Ubuntu project | Ubuntu", Ubuntu.com. [Online]. Available: https://www.ubuntu.com/about. [Accessed: 30- Apr- 2017].

[58] "Windows XP", En.wikipedia.org. [Online]. Available: https://en.wikipedia.org/wiki/Windows_XP#Service_Pack_3. [Accessed: 30- Apr- 2017].

[59]B. Linkletter, "Open-Source Network Simulators", Open-Source Routing and Network Simulation, 2017. [Online]. Available: http://www.brianlinkletter.com/open-source-network-simulators/. [Accessed: 24- Apr- 2017].

[60]"Rochester Institute of Technology - IST/CSEC is OnTheHub", OnTheHub.com, 2016. [Online]. Available: https://e5.onthehub.com/WebStore/ProductsByMajorVersionList.aspx?cmi_cs=1&cmi_mnuMain=16a020b5-ed3c-df11-b4ab-0030487d8897&ws=a7005a5b-58ca-de11-baeb-0030487d8896&vsro=8. [Accessed: 20- Sep- 2016].

[61]"Download Software - Support - Juniper Networks", Juniper.net. [Online]. Available: http://www.juniper.net/support/downloads/?p=firefly#sw. [Accessed: 20- Sep- 2016].

[62]P. Quantin, "SampleCaptures - The Wireshark Wiki", Wiki.wireshark.org. [Online]. Available: https://wiki.wireshark.org/SampleCaptures#SIP_and_RTP. [Accessed: 21- Sep- 2016].

[63] 2.bp.blogspot.com. [Online]. Available: http://2.bp.blogspot.com/_zbwVK_dCFJQ/S0F7e6HjL8I/AAAAAAAACUE/_ob5xQyq2GU/s640/image004.jpg. [Accessed: 26- Apr- 2017].

8 Appendix

This section includes the configurations of all virtual routers.

PE-1

set version 12.1X47-D15.4

set system host-name PE-1

set system root-authentication encrypted-password

"1HbwHNy.o$o8YQRduNh3HniuhX01WDo."

set system login user utkarsh full-name utkarsh

set system login user utkarsh uid 2001

set system login user utkarsh class super-user

set system login user utkarsh authentication encrypted-password

"1.6Rss6QQ$FKs7U/w3At08dWMLMdeWh/"

set system services ssh

set system services web-management http interface ge-0/0/0.0

set system syslog user * any emergency

set system syslog file messages any any

set system syslog file messages authorization info

set system syslog file interactive-commands interactive-commands any

set system license autoupdate url https://ae1.juniper.net/junos/key_retrieval

set interfaces ge-0/0/0 unit 0 family inet address 10.210.30.2/24

set interfaces ge-0/0/1 speed 100m

set interfaces ge-0/0/1 link-mode full-duplex

set interfaces ge-0/0/1 gigether-options no-auto-negotiation

set interfaces ge-0/0/1 unit 0 family inet filter input rpm-classifier

set interfaces ge-0/0/1 unit 0 family inet address 40.40.40.2/30

set interfaces ge-0/0/2 speed 100m

set interfaces ge-0/0/2 link-mode full-duplex

set interfaces ge-0/0/2 gigether-options no-auto-negotiation

set interfaces ge-0/0/2 unit 0 family inet address 172.32.16.1/30

set interfaces ge-0/0/2 unit 0 family mpls

set interfaces ge-0/0/3 speed 100m

set interfaces ge-0/0/3 link-mode full-duplex

set interfaces ge-0/0/3 gigether-options no-auto-negotiation

set interfaces ge-0/0/3 unit 0 family inet address 172.32.16.21/30

set interfaces ge-0/0/3 unit 0 family mpls

set interfaces ge-0/0/4 speed 100m

set interfaces ge-0/0/4 link-mode full-duplex

set interfaces ge-0/0/4 gigether-options no-auto-negotiation

set interfaces ge-0/0/4 unit 0 family inet address 172.32.16.5/30

set interfaces ge-0/0/4 unit 0 family mpls

set interfaces lo0 unit 0 family inet address 172.32.255.100/32

set interfaces lo0 unit 0 family mpls

set routing-options router-id 172.32.255.100

set routing-options autonomous-system 200

set routing-options forwarding-table export load-bal

deactivate routing-options forwarding-table

set protocols rsvp interface all link-protection

deactivate protocols rsvp interface all link-protection

set protocols mpls auto-policing class all drop

set protocols mpls icmp-tunneling

set protocols mpls no-cspf

set protocols mpls oam bfd-liveness-detection minimum-interval 300

set protocols mpls oam bfd-liveness-detection multiplier 2

set protocols mpls oam bfd-liveness-detection failure-action teardown

deactivate protocols mpls oam

set protocols mpls label-switched-path PE-1-to-PE-2 to 172.32.255.200

set protocols mpls label-switched-path PE-1-to-PE-2 bandwidth 10m

set protocols mpls label-switched-path PE-1-to-PE-2 fast-reroute hop-limit 15

deactivate protocols mpls label-switched-path PE-1-to-PE-2 fast-reroute

set protocols mpls label-switched-path PE-1-to-PE-2 primary Best_Path

set protocols mpls label-switched-path PE-1-to-PE-2 secondary Failover_Path standby

deactivate protocols mpls label-switched-path PE-1-to-PE-2 secondary Failover_Path standby

set protocols mpls path Best_Path_cspf 172.32.16.6 strict

set protocols mpls path Best_Path_cspf 172.32.255.7 strict

set protocols mpls path Failover_Path_cspf 172.32.16.2 strict

set protocols mpls path Failover_Path_cspf 172.32.255.3 strict

set protocols mpls path Failover_Path_cspf 172.32.255.6 strict

set protocols mpls path Failover_Path_cspf 172.32.255.5 strict

set protocols mpls path Failover_Path_cspf 172.32.255.8 strict

set protocols mpls path Failover_Path_cspf 172.32.255.9 strict

set protocols mpls path Failover_Path_cspf 172.32.255.10 strict

set protocols mpls path Best_Path 172.32.16.6 strict

set protocols mpls path Best_Path 172.32.16.34 strict

set protocols mpls path Best_Path 172.32.16.90 strict

set protocols mpls path Failover_Path 172.32.16.2 strict

set protocols mpls path Failover_Path 172.32.16.14 strict

set protocols mpls path Failover_Path 172.32.16.18 strict

set protocols mpls path Failover_Path 172.32.16.50 strict

set protocols mpls path Failover_Path 172.32.16.54 strict

set protocols mpls path Failover_Path 172.32.16.62 strict

set protocols mpls path Failover_Path 172.32.16.78 strict

set protocols mpls path Failover_Path 172.32.16.82 strict

set protocols mpls interface all

set protocols bgp group internal type internal

set protocols bgp group internal local-address 172.32.255.100

set protocols bgp group internal family inet unicast

set protocols bgp group internal family inet-vpn unicast

set protocols bgp group internal export nhs

set protocols bgp group internal neighbor 172.32.255.200

set protocols ospf traffic-engineering

deactivate protocols ospf traffic-engineering

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/4.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/4.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface lo0.0

set policy-options policy-statement load-bal then load-balance per-packet

set policy-options policy-statement nhs then next-hop self

set policy-options policy-statement vrf-export-policy term 1 from protocol bgp

set policy-options policy-statement vrf-export-policy term 1 from protocol direct

set policy-options policy-statement vrf-export-policy term 1 then community add customer-a

set policy-options policy-statement vrf-export-policy term 1 then accept

set policy-options policy-statement vrf-export-policy term 2 then reject

set policy-options policy-statement vrf-import-policy term 1 from protocol bgp

set policy-options policy-statement vrf-import-policy term 1 from community customer-a

set policy-options policy-statement vrf-import-policy term 1 then accept

set policy-options policy-statement vrf-import-policy term 2 then reject

set policy-options community customer-a members target:100:200

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service forwarding-classes queue 0 DATA_BE

set class-of-service forwarding-classes queue 0 priority low

set class-of-service forwarding-classes queue 1 VIDEO_AF

set class-of-service forwarding-classes queue 1 priority low

set class-of-service forwarding-classes queue 2 VOICE_EF

set class-of-service forwarding-classes queue 2 priority high

set class-of-service forwarding-classes queue 3 NETWORK_CONTROL

set class-of-service forwarding-classes queue 3 priority high

set class-of-service forwarding-classes queue 4 test

set class-of-service forwarding-classes queue 4 priority high

set class-of-service interfaces ge-* scheduler-map traffic-scheduler-map

set class-of-service interfaces ge-* unit * classifiers exp traffic_classifier_p_devices

set class-of-service interfaces ge-* unit * rewrite-rules exp traffic_rewrite_rule

set class-of-service interfaces ge-0/0/1 scheduler-map traffic-scheduler-map

set class-of-service interfaces ge-0/0/1 unit 0 classifiers inet-precedence traffic_classifier_pe_devices

set class-of-service interfaces ge-0/0/1 unit 0 rewrite-rules exp traffic_rewrite_rule

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class DATA_BE loss-priority low code-point 000

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VIDEO_AF loss-priority low code-point 010

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class NETWORK_CONTROL loss-priority low code-point 110

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VOICE_EF loss-priority low code-point 101

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class DATA_BE scheduler data

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VOICE_EF scheduler voice

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VIDEO_AF scheduler video

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class NETWORK_CONTROL scheduler network

set class-of-service schedulers voice transmit-rate percent 15

set class-of-service schedulers voice buffer-size percent 0

set class-of-service schedulers voice priority high

set class-of-service schedulers video transmit-rate percent 40

set class-of-service schedulers video buffer-size percent 40

set class-of-service schedulers video priority low

set class-of-service schedulers network transmit-rate percent 10

set class-of-service schedulers network buffer-size percent 10

set class-of-service schedulers network priority high

set class-of-service schedulers data transmit-rate remainder

set class-of-service schedulers data buffer-size remainder

set class-of-service schedulers data priority low

set security forwarding-options family mpls mode packet-based

set firewall family inet filter rpm-classifier term rpm-video-classifier from dscp af11

set firewall family inet filter rpm-classifier term rpm-video-classifier then count rpm-video-classifier-counter

set firewall family inet filter rpm-classifier term rpm-video-classifier then forwarding-class VIDEO_AF

set firewall family inet filter rpm-classifier term rpm-video-classifier then accept

set firewall family inet filter rpm-classifier term video-traffic-classifier from destination-port 5004

set firewall family inet filter rpm-classifier term video-traffic-classifier then count video-traffic-classifier-counter

set firewall family inet filter rpm-classifier term video-traffic-classifier then forwarding-class VIDEO_AF

set firewall family inet filter rpm-classifier term video-traffic-classifier then accept

set firewall family inet filter rpm-classifier term rpm-voice-classifier from dscp ef

set firewall family inet filter rpm-classifier term rpm-voice-classifier then count rpm-voice-classifier-counter

set firewall family inet filter rpm-classifier term rpm-voice-classifier then forwarding-class VOICE_EF

set firewall family inet filter rpm-classifier term rpm-voice-classifier then accept

set firewall family inet filter rpm-classifier term voice-traffic-classifer from precedence critical-ecp

set firewall family inet filter rpm-classifier term voice-traffic-classifer then count new-voice-2017-counter

set firewall family inet filter rpm-classifier term voice-traffic-classifer then forwarding-class VOICE_EF

set firewall family inet filter rpm-classifier term voice-traffic-classifer then accept

set firewall family inet filter rpm-classifier term data-rpm-and-traffic-classifier then count data-rpm-and-traffic-counter

set firewall family inet filter rpm-classifier term data-rpm-and-traffic-classifier then forwarding-class DATA_BE

set firewall family inet filter rpm-classifier term data-rpm-and-traffic-classifier then accept

set routing-instances Customer-A instance-type vrf

set routing-instances Customer-A interface ge-0/0/1.0

set routing-instances Customer-A route-distinguisher 172.32.255.100:1

set routing-instances Customer-A vrf-import vrf-import-policy

set routing-instances Customer-A vrf-export vrf-export-policy

set routing-instances Customer-A vrf-table-label

set routing-instances Customer-A protocols bgp group external type external

set routing-instances Customer-A protocols bgp group external peer-as 100

set routing-instances Customer-A protocols bgp group external as-override

set routing-instances Customer-A protocols bgp group external neighbor 40.40.40.1

PE-2

set version 12.1X47-D15.4

```
set system host-name PE-2
set system root-authentication encrypted-password
"$1$HbwHNy.o$o8YQRduNh3HniuhX01WDo."
set system login user utkarsh full-name utkarsh
set system login user utkarsh uid 2000
set system login user utkarsh class super-user
set system login user utkarsh authentication encrypted-password
"$1$.6Rss6QQ$FKs7U/w3At08dWMLMdeWh/"
set system services ssh
set system services web-management http interface ge-0/0/0.0
set system syslog user * any emergency
set system syslog file messages any any
set system syslog file messages authorization info
set system syslog file interactive-commands interactive-commands any
set system license autoupdate url https://ae1.juniper.net/junos/key_retrieval
set interfaces ge-0/0/0 unit 0 family inet address 10.210.30.3/24
set interfaces ge-0/0/1 speed 100m
set interfaces ge-0/0/1 link-mode full-duplex
set interfaces ge-0/0/1 gigether-options no-auto-negotiation
set interfaces ge-0/0/1 unit 0 family inet address 40.40.40.5/30
set interfaces ge-0/0/2 speed 100m
set interfaces ge-0/0/2 link-mode full-duplex
set interfaces ge-0/0/2 gigether-options no-auto-negotiation
set interfaces ge-0/0/2 unit 0 family inet address 172.32.16.74/30
set interfaces ge-0/0/2 unit 0 family mpls
```

set interfaces ge-0/0/3 speed 100m

set interfaces ge-0/0/3 link-mode full-duplex

set interfaces ge-0/0/3 gigether-options no-auto-negotiation

set interfaces ge-0/0/3 unit 0 family inet address 172.32.16.82/30

set interfaces ge-0/0/3 unit 0 family mpls

set interfaces ge-0/0/4 speed 100m

set interfaces ge-0/0/4 link-mode full-duplex

set interfaces ge-0/0/4 gigether-options no-auto-negotiation

set interfaces ge-0/0/4 unit 0 family inet address 172.32.16.90/30

set interfaces ge-0/0/4 unit 0 family mpls

set interfaces lo0 unit 0 family inet address 172.32.255.200/32

set interfaces lo0 unit 0 family mpls

set routing-options router-id 172.32.255.200

set routing-options autonomous-system 200

set routing-options forwarding-table export load-bal

deactivate routing-options forwarding-table

set protocols rsvp interface all link-protection

deactivate protocols rsvp interface all link-protection

set protocols mpls auto-policing class all drop

set protocols mpls icmp-tunneling

set protocols mpls no-cspf

set protocols mpls oam bfd-liveness-detection minimum-interval 300

set protocols mpls oam bfd-liveness-detection multiplier 2

set protocols mpls oam bfd-liveness-detection failure-action teardown

deactivate protocols mpls oam

set protocols mpls label-switched-path PE-2-to-PE-1 to 172.32.255.100

set protocols mpls label-switched-path PE-2-to-PE-1 bandwidth 10m

set protocols mpls label-switched-path PE-2-to-PE-1 fast-reroute hop-limit 15

deactivate protocols mpls label-switched-path PE-2-to-PE-1 fast-reroute

set protocols mpls label-switched-path PE-2-to-PE-1 primary Best_Path

set protocols mpls label-switched-path PE-2-to-PE-1 secondary Failover_Path standby

deactivate protocols mpls label-switched-path PE-2-to-PE-1 secondary Failover_Path standby

set protocols mpls path Best_Path_cspf 172.32.16.89 strict

set protocols mpls path Best_Path_cspf 172.32.255.7

set protocols mpls path Best_Path_cspf 172.32.255.2

set protocols mpls path Failover_Path_cspf 172.32.16.81 strict

set protocols mpls path Failover_Path_cspf 172.32.255.10 strict

set protocols mpls path Failover_Path_cspf 172.32.255.9 strict

set protocols mpls path Failover_Path_cspf 172.32.255.8 strict

set protocols mpls path Failover_Path_cspf 172.32.255.5 strict

set protocols mpls path Failover_Path_cspf 172.32.255.6 strict

set protocols mpls path Failover_Path_cspf 172.32.255.3 strict

set protocols mpls path Failover_Path_cspf 172.32.255.1 strict

set protocols mpls path Best_Path 172.32.16.89 strict

set protocols mpls path Best_Path 172.32.16.33 strict

set protocols mpls path Best_Path 172.32.16.5 strict

set protocols mpls path Failover_Path 172.32.16.81 strict

set protocols mpls path Failover_Path 172.32.16.77 strict

set protocols mpls path Failover_Path 172.32.16.61 strict

set protocols mpls path Failover_Path 172.32.16.53 strict

set protocols mpls path Failover_Path 172.32.16.49 strict

set protocols mpls path Failover_Path 172.32.16.17 strict

set protocols mpls path Failover_Path 172.32.16.13 strict

set protocols mpls path Failover_Path 172.32.16.1 strict

set protocols mpls interface all

set protocols bgp group internal type internal

set protocols bgp group internal local-address 172.32.255.200

set protocols bgp group internal family inet unicast

set protocols bgp group internal family inet-vpn unicast

set protocols bgp group internal export nhs

set protocols bgp group internal neighbor 172.32.255.100

set protocols ospf traffic-engineering

deactivate protocols ospf traffic-engineering

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 bfd-liveness-detection minimum-interval

1200

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 bfd-liveness-detection minimum-interval

1200

set protocols ospf area 0.0.0.0 interface ge-0/0/4.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/4.0 bfd-liveness-detection minimum-interval

1200

set protocols ospf area 0.0.0.0 interface lo0.0

set policy-options policy-statement load-bal then load-balance per-packet

set policy-options policy-statement nhs then next-hop self

set policy-options policy-statement vrf-export-policy term 1 from protocol bgp

set policy-options policy-statement vrf-export-policy term 1 from protocol direct

set policy-options policy-statement vrf-export-policy term 1 then community add customer-a

set policy-options policy-statement vrf-export-policy term 1 then accept

set policy-options policy-statement vrf-export-policy term 2 then reject

set policy-options policy-statement vrf-import-policy term 1 from protocol bgp

set policy-options policy-statement vrf-import-policy term 1 from community customer-a

set policy-options policy-statement vrf-import-policy term 1 then accept

set policy-options policy-statement vrf-import-policy term 2 then reject

set policy-options community customer-a members target:100:200

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service forwarding-classes queue 0 DATA_BE

set class-of-service forwarding-classes queue 0 priority low

set class-of-service forwarding-classes queue 1 VIDEO_AF

set class-of-service forwarding-classes queue 1 priority low

set class-of-service forwarding-classes queue 2 VOICE_EF

set class-of-service forwarding-classes queue 2 priority high

set class-of-service forwarding-classes queue 3 NETWORK_CONTROL

set class-of-service forwarding-classes queue 3 priority high

set class-of-service forwarding-classes queue 4 test

set class-of-service forwarding-classes queue 4 priority high

set class-of-service interfaces ge-* scheduler-map traffic-scheduler-map

set class-of-service interfaces ge-* unit * classifiers exp traffic_classifier_p_devices

set class-of-service interfaces ge-* unit * rewrite-rules exp traffic_rewrite_rule

set class-of-service interfaces ge-0/0/1 scheduler-map traffic-scheduler-map

set class-of-service interfaces ge-0/0/1 unit 0 classifiers inet-precedence traffic_classifier_pe_devices

set class-of-service interfaces ge-0/0/1 unit 0 rewrite-rules exp traffic_rewrite_rule

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class DATA_BE loss-priority low code-point 000

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VIDEO_AF loss-priority low code-point 010

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class NETWORK_CONTROL loss-priority low code-point 110

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VOICE_EF loss-priority low code-point 101

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class DATA_BE scheduler data

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VOICE_EF scheduler voice

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VIDEO_AF scheduler video

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class NETWORK_CONTROL scheduler network

set class-of-service schedulers voice transmit-rate percent 15

set class-of-service schedulers voice buffer-size percent 0

set class-of-service schedulers voice priority high

set class-of-service schedulers video transmit-rate percent 40

set class-of-service schedulers video buffer-size percent 40

set class-of-service schedulers video priority low

set class-of-service schedulers network transmit-rate percent 10

set class-of-service schedulers network buffer-size percent 10

set class-of-service schedulers network priority high

set class-of-service schedulers data transmit-rate remainder

set class-of-service schedulers data buffer-size remainder

set class-of-service schedulers data priority low

set security forwarding-options family mpls mode packet-based

set routing-instances CE-A instance-type vrf

set routing-instances CE-A interface ge-0/0/1.0

set routing-instances CE-A route-distinguisher 172.32.255.200:1

set routing-instances CE-A vrf-import vrf-import-policy

set routing-instances CE-A vrf-export vrf-export-policy

set routing-instances CE-A vrf-table-label

set routing-instances CE-A protocols bgp group external type external

set routing-instances CE-A protocols bgp group external traceoptions file bgp-not-working

set routing-instances CE-A protocols bgp group external traceoptions flag open detail

set routing-instances CE-A protocols bgp group external traceoptions flag packets detail

set routing-instances CE-A protocols bgp group external peer-as 100

set routing-instances CE-A protocols bgp group external as-override

set routing-instances CE-A protocols bgp group external neighbor 40.40.40.6

Customer-A-Site-1

set version 12.1X47-D15.4

set groups eth-speed

set system host-name Customer-A-Site-1

set system root-authentication encrypted-password

"$1$7f6kjzRZ$jyaz5mU2vTuAeguzvVHR80"

set system login user utkarsh full-name utkarsh

set system login user utkarsh uid 2001

set system login user utkarsh class super-user

set system login user utkarsh authentication encrypted-password

"1.6Rss6QQ$FKs7U/w3At08dWMLMdeWh/"

set system services ftp

set system services ssh

set system services telnet

set system services web-management http interface ge-0/0/0.0

set system syslog user * any emergency

set system syslog file messages any any

set system syslog file messages authorization info

set system syslog file interactive-commands interactive-commands any

set system license autoupdate url https://ae1.juniper.net/junos/key_retrieval

set interfaces ge-0/0/0 unit 0 family inet address 10.210.30.7/24

set interfaces ge-0/0/1 unit 0 family inet address 40.40.40.1/30

set interfaces ge-0/0/2 mac 00:0c:29:f4:32:f7

set interfaces ge-0/0/2 unit 0 family inet filter input congested-voice

set interfaces ge-0/0/2 unit 0 family inet policer input congested-voice

deactivate interfaces ge-0/0/2 unit 0 family inet policer

set interfaces ge-0/0/2 unit 0 family inet address 10.10.10.1/30

set interfaces ge-0/0/3 enable

set interfaces ge-0/0/3 unit 0 family inet filter input congested-dav

set interfaces ge-0/0/3 unit 0 family inet address 10.10.20.1/30

set interfaces lo0 unit 0 family inet address 40.40.255.1/32

set routing-options static route 50.50.50.50/32 receive

set routing-options static route 0.0.0.0/0 next-hop 40.40.40.2

set routing-options router-id 40.40.255.1

set routing-options autonomous-system 100

set protocols bgp group external type external

set protocols bgp group external export test-route

set protocols bgp group external peer-as 200

set protocols bgp group external neighbor 40.40.40.2

set policy-options policy-statement test-route term 1 from protocol static

set policy-options policy-statement test-route term 1 from protocol direct

set policy-options policy-statement test-route term 1 then accept

set class-of-service classifiers inet-precedence traffic_classifier forwarding-class DATA_BE

loss-priority low code-points 000

set class-of-service classifiers inet-precedence traffic_classifier forwarding-class VIDEO_AF

loss-priority low code-points 010

set class-of-service classifiers inet-precedence traffic_classifier forwarding-class VOICE_EF

loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier forwarding-class

NETWORK_CONTROL loss-priority low code-points 110

set class-of-service forwarding-classes queue 0 DATA_BE

set class-of-service forwarding-classes queue 0 priority low

set class-of-service forwarding-classes queue 1 VIDEO_AF

set class-of-service forwarding-classes queue 1 priority low

set class-of-service forwarding-classes queue 2 VOICE_EF

set class-of-service forwarding-classes queue 2 priority high

set class-of-service forwarding-classes queue 3 NETWORK_CONTROL

set class-of-service forwarding-classes queue 3 priority high

set class-of-service interfaces ge-* scheduler-map traffic-scheduler-map

set class-of-service interfaces ge-* unit * classifiers inet-precedence traffic_classifier

set class-of-service interfaces ge-* unit * rewrite-rules inet-precedence traffic_rewrite_rule

set class-of-service rewrite-rules inet-precedence traffic_rewrite_rule forwarding-class DATA_BE loss-priority low code-point 000

set class-of-service rewrite-rules inet-precedence traffic_rewrite_rule forwarding-class VIDEO_AF loss-priority low code-point 010

set class-of-service rewrite-rules inet-precedence traffic_rewrite_rule forwarding-class VOICE_EF loss-priority low code-point 101

set class-of-service rewrite-rules inet-precedence traffic_rewrite_rule forwarding-class NETWORK_CONTROL loss-priority low code-point 110

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class DATA_BE scheduler data

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VOICE_EF scheduler voice

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VIDEO_AF scheduler video

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class NETWORK_CONTROL scheduler network

set class-of-service schedulers voice transmit-rate percent 15

set class-of-service schedulers voice buffer-size percent 0

set class-of-service schedulers voice priority high

set class-of-service schedulers video transmit-rate percent 60

set class-of-service schedulers video buffer-size percent 60

set class-of-service schedulers video priority low

set class-of-service schedulers network transmit-rate percent 10

set class-of-service schedulers network buffer-size percent 10

set class-of-service schedulers network priority high

set class-of-service schedulers data transmit-rate remainder

set class-of-service schedulers data buffer-size remainder

set class-of-service schedulers data priority low

set security forwarding-options family mpls mode packet-based

set firewall family inet filter normal-dav term 1 from port 5004

set firewall family inet filter normal-dav term 1 then policer normal-video

set firewall family inet filter normal-dav term 1 then count video-in-profile

set firewall family inet filter normal-dav term 1 then forwarding-class VIDEO_AF

set firewall family inet filter normal-dav term 1 then accept

set firewall family inet filter normal-dav term 2 then policer normal-data

set firewall family inet filter normal-dav term 2 then count data-in-profile

set firewall family inet filter normal-dav term 2 then forwarding-class DATA_BE

set firewall family inet filter normal-dav term 2 then accept

set firewall family inet filter normal-voice term 1 then policer normal-voice

set firewall family inet filter normal-voice term 1 then count voice-in-profile

set firewall family inet filter normal-voice term 1 then forwarding-class VOICE_EF

set firewall family inet filter normal-voice term 1 then accept

set firewall family inet filter congested-dav term 1 from port 5004

set firewall family inet filter congested-dav term 1 then policer congested-video

set firewall family inet filter congested-dav term 1 then count video-in-profile

set firewall family inet filter congested-dav term 1 then forwarding-class VIDEO_AF

set firewall family inet filter congested-dav term 1 then accept

set firewall family inet filter congested-dav term 2 then policer congested-data

set firewall family inet filter congested-dav term 2 then count data-in-profile

set firewall family inet filter congested-dav term 2 then forwarding-class DATA_BE

set firewall family inet filter congested-dav term 2 then accept

set firewall family inet filter congested-voice term 1 then policer congested-voice

set firewall family inet filter congested-voice term 1 then count voice-in-profile

set firewall family inet filter congested-voice term 1 then forwarding-class VOICE_EF

set firewall family inet filter congested-voice term 1 then accept

set firewall policer normal-video if-exceeding bandwidth-limit 4m

set firewall policer normal-video if-exceeding burst-size-limit 15k

set firewall policer normal-video then discard

set firewall policer congested-video if-exceeding bandwidth-limit 6m

set firewall policer congested-video if-exceeding burst-size-limit 15k

set firewall policer congested-video then discard

set firewall policer normal-voice if-exceeding bandwidth-limit 2m

set firewall policer normal-voice if-exceeding burst-size-limit 15k

set firewall policer normal-voice then discard

set firewall policer congested-voice if-exceeding bandwidth-limit 3m

set firewall policer congested-voice if-exceeding burst-size-limit 15k

set firewall policer congested-voice then discard

set firewall policer normal-data if-exceeding bandwidth-limit 2m

set firewall policer normal-data if-exceeding burst-size-limit 15k

set firewall policer normal-data then discard

set firewall policer congested-data if-exceeding bandwidth-limit 3m

set firewall policer congested-data if-exceeding burst-size-limit 15k

set firewall policer congested-data then discard

set services rpm probe Utkarsh test video_test probe-type icmp-ping-timestamp

set services rpm probe Utkarsh test video_test target address 40.40.40.6

set services rpm probe Utkarsh test video_test probe-count 15

set services rpm probe Utkarsh test video_test probe-interval 1

set services rpm probe Utkarsh test video_test test-interval 1

set services rpm probe Utkarsh test video_test source-address 40.40.40.1

set services rpm probe Utkarsh test video_test history-size 600

set services rpm probe Utkarsh test video_test dscp-code-points af11

set services rpm probe Utkarsh test video_test data-size 1370

deactivate services rpm probe Utkarsh test video_test

set services rpm probe Utkarsh test voice_test probe-type icmp-ping-timestamp

set services rpm probe Utkarsh test voice_test target address 40.40.40.6

set services rpm probe Utkarsh test voice_test probe-count 15

set services rpm probe Utkarsh test voice_test probe-interval 1

set services rpm probe Utkarsh test voice_test test-interval 1

set services rpm probe Utkarsh test voice_test source-address 40.40.40.1

set services rpm probe Utkarsh test voice_test history-size 600

set services rpm probe Utkarsh test voice_test dscp-code-points ef

set services rpm probe Utkarsh test voice_test data-size 214

deactivate services rpm probe Utkarsh test voice_test

set services rpm probe Utkarsh test data_test probe-type icmp-ping-timestamp

set services rpm probe Utkarsh test data_test target address 40.40.40.6

set services rpm probe Utkarsh test data_test probe-count 15

set services rpm probe Utkarsh test data_test probe-interval 1

set services rpm probe Utkarsh test data_test test-interval 1

set services rpm probe Utkarsh test data_test source-address 40.40.40.1

set services rpm probe Utkarsh test data_test history-size 600

set services rpm probe Utkarsh test data_test dscp-code-points be

set services rpm probe Utkarsh test data_test data-size 1436

set services rpm probe Utkarsh test data_test hardware-timestamp

Customer-A-Site-2

set version 12.1X47-D15.4

set system host-name Site-2

set system root-authentication encrypted-password

"1GQtlaQSc$xhd6oMzDiN4j8vTWynIdx/"

set system login user utkarsh full-name utkarsh

set system login user utkarsh uid 2000

set system login user utkarsh class super-user

set system login user utkarsh authentication encrypted-password

"1.6Rss6QQ$FKs7U/w3At08dWMLMdeWh/"

set system services ftp

set system services ssh

set system services telnet

set system services web-management http interface ge-0/0/0.0

set system syslog user * any emergency

set system syslog file messages any any

set system syslog file messages authorization info

set system syslog file interactive-commands interactive-commands any

set system license autoupdate url https://ae1.juniper.net/junos/key_retrieval

set interfaces ge-0/0/0 unit 0 family inet address 10.210.30.8/24

set interfaces ge-0/0/1 unit 0 family inet address 40.40.40.6/30

set interfaces ge-0/0/3 unit 0 family inet address 192.168.1.1/30

set interfaces lo0 unit 0 family inet address 20.20.255.2/32

set routing-options static route 60.60.60.60/32 receive

set routing-options static route 0.0.0.0/0 receive

set routing-options router-id 40.40.255.2

set routing-options autonomous-system 50

set protocols bgp group external type external

set protocols bgp group external export test-route

set protocols bgp group external peer-as 200

set protocols bgp group external neighbor 40.40.40.5

set policy-options policy-statement test-route term 1 from protocol static

set policy-options policy-statement test-route term 1 from protocol direct

set policy-options policy-statement test-route term 1 then accept

set security forwarding-options family mpls mode packet-based

P-1

set version 12.1X47-D15.4

set system host-name P-1

set system root-authentication encrypted-password

"1HbwHNy.o$o8YQRduNh3HniuhX01WDo."

set system login user utkarsh full-name utkarsh

set system login user utkarsh uid 2000

set system login user utkarsh class super-user

set system login user utkarsh authentication encrypted-password

"1.6Rss6QQ$FKs7U/w3At08dWMLMdeWh/"

set system services ssh

set system services web-management http interface ge-0/0/0.0

set system syslog user * any emergency

set system syslog file messages any any

set system syslog file messages authorization info

set system syslog file interactive-commands interactive-commands any

set system license autoupdate url https://ae1.juniper.net/junos/key_retrieval

set interfaces ge-0/0/0 unit 0 family inet address 10.210.30.10/24

set interfaces ge-0/0/1 speed 100m

set interfaces ge-0/0/1 link-mode full-duplex

set interfaces ge-0/0/1 gigether-options no-auto-negotiation

set interfaces ge-0/0/1 unit 0 family inet address 172.32.16.2/30

set interfaces ge-0/0/1 unit 0 family mpls

set interfaces ge-0/0/2 speed 100m

set interfaces ge-0/0/2 link-mode full-duplex

set interfaces ge-0/0/2 gigether-options no-auto-negotiation

set interfaces ge-0/0/2 unit 0 family inet address 172.32.16.9/30

set interfaces ge-0/0/2 unit 0 family mpls

set interfaces ge-0/0/3 speed 100m

set interfaces ge-0/0/3 link-mode full-duplex

set interfaces ge-0/0/3 gigether-options no-auto-negotiation

set interfaces ge-0/0/3 unit 0 family inet address 172.32.16.13/30

set interfaces ge-0/0/3 unit 0 family mpls

set interfaces ge-0/0/4 speed 100m

set interfaces ge-0/0/4 link-mode full-duplex

set interfaces ge-0/0/4 gigether-options no-auto-negotiation

set interfaces lo0 unit 0 family inet address 172.32.255.1/32

set interfaces lo0 unit 0 family mpls

set routing-options router-id 172.32.255.1

set routing-options autonomous-system 200

set protocols rsvp interface all link-protection

deactivate protocols rsvp interface all link-protection

set protocols mpls icmp-tunneling

set protocols mpls no-cspf

set protocols mpls interface all

set protocols ospf traffic-engineering

deactivate protocols ospf traffic-engineering

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 bfd-liveness-detection minimum-interval

1200

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface lo0.0

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service forwarding-classes queue 0 DATA_BE

set class-of-service forwarding-classes queue 0 priority low

set class-of-service forwarding-classes queue 1 VIDEO_AF

set class-of-service forwarding-classes queue 1 priority low

set class-of-service forwarding-classes queue 2 VOICE_EF

set class-of-service forwarding-classes queue 2 priority high

set class-of-service forwarding-classes queue 3 NETWORK_CONTROL

set class-of-service forwarding-classes queue 3 priority high

set class-of-service forwarding-classes queue 4 test

set class-of-service forwarding-classes queue 4 priority high

set class-of-service interfaces ge-* scheduler-map traffic-scheduler-map

set class-of-service interfaces ge-* unit * classifiers exp traffic_classifier_p_devices

set class-of-service interfaces ge-* unit * rewrite-rules exp traffic_rewrite_rule

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class DATA_BE loss-priority low code-point 000

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VIDEO_AF loss-priority low code-point 010

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class NETWORK_CONTROL loss-priority low code-point 110

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VOICE_EF loss-priority low code-point 101

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class DATA_BE scheduler data

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VOICE_EF scheduler voice

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VIDEO_AF scheduler video

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class NETWORK_CONTROL scheduler network

set class-of-service schedulers voice transmit-rate percent 15

set class-of-service schedulers voice buffer-size percent 0

set class-of-service schedulers voice priority high

set class-of-service schedulers video transmit-rate percent 40

set class-of-service schedulers video buffer-size percent 40

set class-of-service schedulers video priority low

set class-of-service schedulers network transmit-rate percent 10

set class-of-service schedulers network buffer-size percent 10

set class-of-service schedulers network priority high

set class-of-service schedulers data transmit-rate remainder

set class-of-service schedulers data buffer-size remainder

set class-of-service schedulers data priority low

set security forwarding-options family mpls mode packet-based

P-2

set version 12.1X47-D15.4

set system host-name P-2

set system root-authentication encrypted-password

"1HbwHNy.o$o8YQRduNh3HniuhX01WDo."

set system login user utkarsh full-name utkarsh

set system login user utkarsh uid 2001

set system login user utkarsh class super-user

set system login user utkarsh authentication encrypted-password

"1.6Rss6QQ$FKs7U/w3At08dWMLMdeWh/"

set system services ssh

set system services web-management http interface ge-0/0/0.0

set system syslog user * any emergency

set system syslog file messages any any

set system syslog file messages authorization info

set system syslog file interactive-commands interactive-commands any

set system license autoupdate url https://ae1.juniper.net/junos/key_retrieval

set interfaces ge-0/0/0 unit 0 family inet address 10.210.30.11/24

set interfaces ge-0/0/1 speed 100m

set interfaces ge-0/0/1 link-mode full-duplex

set interfaces ge-0/0/1 gigether-options no-auto-negotiation

set interfaces ge-0/0/1 unit 0 family inet address 172.32.16.6/30

set interfaces ge-0/0/1 unit 0 family mpls

set interfaces ge-0/0/2 speed 100m

set interfaces ge-0/0/2 link-mode full-duplex

set interfaces ge-0/0/2 gigether-options no-auto-negotiation

set interfaces ge-0/0/2 unit 0 family inet address 172.32.16.10/30

set interfaces ge-0/0/2 unit 0 family mpls

set interfaces ge-0/0/3 speed 100m

set interfaces ge-0/0/3 link-mode full-duplex

set interfaces ge-0/0/3 gigether-options no-auto-negotiation

set interfaces ge-0/0/3 unit 0 family inet address 172.32.16.25/30

set interfaces ge-0/0/3 unit 0 family mpls

set interfaces ge-0/0/4 speed 100m

set interfaces ge-0/0/4 link-mode full-duplex

set interfaces ge-0/0/4 gigether-options no-auto-negotiation

set interfaces ge-0/0/4 unit 0 family inet address 172.32.16.33/30

set interfaces ge-0/0/4 unit 0 family mpls

set interfaces lo0 unit 0 family inet address 172.32.255.2/32

set interfaces lo0 unit 0 family mpls

set routing-options router-id 172.32.255.2

set routing-options autonomous-system 200

set routing-options forwarding-table export load-bal

deactivate routing-options forwarding-table

set protocols rsvp interface all link-protection

deactivate protocols rsvp interface all link-protection

set protocols mpls icmp-tunneling

set protocols mpls no-cspf

set protocols mpls interface all

set protocols ospf traffic-engineering

deactivate protocols ospf traffic-engineering

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 node-link-protection

deactivate protocols ospf area 0.0.0.0 interface ge-0/0/2.0

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 node-link-protection

deactivate protocols ospf area 0.0.0.0 interface ge-0/0/3.0

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 node-link-protection

deactivate protocols ospf area 0.0.0.0 interface ge-0/0/1.0

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/4.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/4.0 node-link-protection

deactivate protocols ospf area 0.0.0.0 interface ge-0/0/4.0

set protocols ospf area 0.0.0.0 interface ge-0/0/4.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface lo0.0

set policy-options policy-statement load-bal then load-balance per-packet

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service forwarding-classes queue 0 DATA_BE

set class-of-service forwarding-classes queue 0 priority low

set class-of-service forwarding-classes queue 1 VIDEO_AF

set class-of-service forwarding-classes queue 1 priority low

set class-of-service forwarding-classes queue 2 VOICE_EF

set class-of-service forwarding-classes queue 2 priority high

set class-of-service forwarding-classes queue 3 NETWORK_CONTROL

set class-of-service forwarding-classes queue 3 priority high

set class-of-service forwarding-classes queue 4 test

set class-of-service forwarding-classes queue 4 priority high

set class-of-service interfaces ge-* scheduler-map traffic-scheduler-map

set class-of-service interfaces ge-* unit * classifiers exp traffic_classifier_p_devices

set class-of-service interfaces ge-* unit * rewrite-rules exp traffic_rewrite_rule

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class DATA_BE loss-priority low code-point 000

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VIDEO_AF loss-priority low code-point 010

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class NETWORK_CONTROL loss-priority low code-point 110

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VOICE_EF loss-priority low code-point 101

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class DATA_BE scheduler data

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VOICE_EF

scheduler voice

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VIDEO_AF

scheduler video

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class

NETWORK_CONTROL scheduler network

set class-of-service schedulers voice transmit-rate percent 15

set class-of-service schedulers voice buffer-size percent 0

set class-of-service schedulers voice priority high

set class-of-service schedulers video transmit-rate percent 40

set class-of-service schedulers video buffer-size percent 40

set class-of-service schedulers video priority low

set class-of-service schedulers network transmit-rate percent 10

set class-of-service schedulers network buffer-size percent 10

set class-of-service schedulers network priority high

set class-of-service schedulers data transmit-rate remainder

set class-of-service schedulers data buffer-size remainder

set class-of-service schedulers data priority low

set security forwarding-options family mpls mode packet-based

P-3

set version 12.1X47-D15.4

set system host-name P-3

```
set system root-authentication encrypted-password
"$1$HbwHNy.o$o8YQRduNh3HniuhX01WDo."
set system login user utkarsh full-name utkarsh
set system login user utkarsh uid 2000
set system login user utkarsh class super-user
set system login user utkarsh authentication encrypted-password
"$1$.6Rss6QQ$FKs7U/w3At08dWMLMdeWh/"
set system services ssh
set system services web-management http interface ge-0/0/0.0
set system syslog user * any emergency
set system syslog file messages any any
set system syslog file messages authorization info
set system syslog file interactive-commands interactive-commands any
set system license autoupdate url https://ae1.juniper.net/junos/key_retrieval
set interfaces ge-0/0/0 unit 0 family inet address 10.210.30.12/24
set interfaces ge-0/0/1 speed 100m
set interfaces ge-0/0/1 link-mode full-duplex
set interfaces ge-0/0/1 gigether-options no-auto-negotiation
set interfaces ge-0/0/1 unit 0 family inet address 172.32.16.14/30
set interfaces ge-0/0/1 unit 0 family mpls
set interfaces ge-0/0/2 speed 100m
set interfaces ge-0/0/2 link-mode full-duplex
set interfaces ge-0/0/2 gigether-options no-auto-negotiation
set interfaces ge-0/0/2 unit 0 family inet address 172.32.16.22/30
set interfaces ge-0/0/2 unit 0 family mpls
```

set interfaces ge-0/0/3 speed 100m

set interfaces ge-0/0/3 link-mode full-duplex

set interfaces ge-0/0/3 gigether-options no-auto-negotiation

set interfaces ge-0/0/3 unit 0 family inet address 172.32.16.26/30

set interfaces ge-0/0/3 unit 0 family mpls

set interfaces ge-0/0/4 speed 100m

set interfaces ge-0/0/4 link-mode full-duplex

set interfaces ge-0/0/4 gigether-options no-auto-negotiation

set interfaces ge-0/0/4 unit 0 family inet address 172.32.16.17/30

set interfaces ge-0/0/4 unit 0 family mpls

set interfaces ge-0/0/5 speed 100m

set interfaces ge-0/0/5 link-mode full-duplex

set interfaces ge-0/0/5 gigether-options no-auto-negotiation

set interfaces ge-0/0/5 unit 0 family inet address 172.32.16.41/30

set interfaces ge-0/0/5 unit 0 family mpls

set interfaces lo0 unit 0 family inet address 172.32.255.3/32

set interfaces lo0 unit 0 family mpls

set routing-options router-id 172.32.255.3

set routing-options autonomous-system 200

set protocols rsvp interface all link-protection

deactivate protocols rsvp interface all link-protection

set protocols mpls icmp-tunneling

set protocols mpls no-cspf

set protocols mpls interface all

set protocols ospf traffic-engineering

deactivate protocols ospf traffic-engineering

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/4.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/4.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/5.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/5.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface lo0.0

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service forwarding-classes queue 0 DATA_BE

set class-of-service forwarding-classes queue 0 priority low

set class-of-service forwarding-classes queue 1 VIDEO_AF

set class-of-service forwarding-classes queue 1 priority low

set class-of-service forwarding-classes queue 2 VOICE_EF

set class-of-service forwarding-classes queue 2 priority high

set class-of-service forwarding-classes queue 3 NETWORK_CONTROL

set class-of-service forwarding-classes queue 3 priority high

set class-of-service forwarding-classes queue 4 test

set class-of-service forwarding-classes queue 4 priority high

set class-of-service interfaces ge-* scheduler-map traffic-scheduler-map

set class-of-service interfaces ge-* unit * classifiers exp traffic_classifier_p_devices

set class-of-service interfaces ge-* unit * rewrite-rules exp traffic_rewrite_rule

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class DATA_BE loss-priority low code-point 000

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VIDEO_AF loss-priority low code-point 010

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class

NETWORK_CONTROL loss-priority low code-point 110

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VOICE_EF loss-

priority low code-point 101

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class DATA_BE

scheduler data

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VOICE_EF

scheduler voice

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VIDEO_AF

scheduler video

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class

NETWORK_CONTROL scheduler network

set class-of-service schedulers voice transmit-rate percent 15

set class-of-service schedulers voice buffer-size percent 0

set class-of-service schedulers voice priority high

set class-of-service schedulers video transmit-rate percent 40

set class-of-service schedulers video buffer-size percent 40

set class-of-service schedulers video priority low

set class-of-service schedulers network transmit-rate percent 10

set class-of-service schedulers network buffer-size percent 10

set class-of-service schedulers network priority high

set class-of-service schedulers data transmit-rate remainder

set class-of-service schedulers data buffer-size remainder

set class-of-service schedulers data priority low

set security forwarding-options family mpls mode packet-based

P-4

set version 12.1X47-D15.4

set system host-name P-4

set system root-authentication encrypted-password

"1HbwHNy.o$o8YQRduNh3HniuhX01WDo."

set system login user utkarsh full-name utkarsh

set system login user utkarsh uid 2000

set system login user utkarsh class super-user

set system login user utkarsh authentication encrypted-password

"1.6Rss6QQ$FKs7U/w3At08dWMLMdeWh/"

set system services ssh

set system services web-management http interface ge-0/0/0.0

set system syslog user * any emergency

set system syslog file messages any any

set system syslog file messages authorization info

set system syslog file interactive-commands interactive-commands any

set system license autoupdate url https://ae1.juniper.net/junos/key_retrieval

set interfaces ge-0/0/0 unit 0 family inet address 10.210.30.13/24

set interfaces ge-0/0/1 speed 100m

set interfaces ge-0/0/1 link-mode full-duplex

set interfaces ge-0/0/1 gigether-options no-auto-negotiation

set interfaces ge-0/0/1 unit 0 family inet address 172.32.16.18/30

set interfaces ge-0/0/1 unit 0 family mpls

set interfaces ge-0/0/2 speed 100m

set interfaces ge-0/0/2 link-mode full-duplex

set interfaces ge-0/0/2 gigether-options no-auto-negotiation

set interfaces ge-0/0/2 unit 0 family inet address 172.32.16.37/30

set interfaces ge-0/0/2 unit 0 family mpls

set interfaces ge-0/0/3 speed 100m

set interfaces ge-0/0/3 link-mode full-duplex

set interfaces ge-0/0/3 gigether-options no-auto-negotiation

set interfaces ge-0/0/3 unit 0 family inet address 172.32.16.49/30

set interfaces ge-0/0/3 unit 0 family mpls

set interfaces lo0 unit 0 family inet address 172.32.255.4/32

set interfaces lo0 unit 0 family mpls

set routing-options router-id 172.32.255.4

set routing-options autonomous-system 200

set protocols rsvp interface all link-protection

deactivate protocols rsvp interface all link-protection

set protocols mpls icmp-tunneling

set protocols mpls no-cspf

set protocols mpls interface all

set protocols ospf traffic-engineering

deactivate protocols ospf traffic-engineering

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 bfd-liveness-detection minimum-interval

1200

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface lo0.0

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service forwarding-classes queue 0 DATA_BE

set class-of-service forwarding-classes queue 0 priority low

set class-of-service forwarding-classes queue 1 VIDEO_AF

set class-of-service forwarding-classes queue 1 priority low

set class-of-service forwarding-classes queue 2 VOICE_EF

set class-of-service forwarding-classes queue 2 priority high

set class-of-service forwarding-classes queue 3 NETWORK_CONTROL

set class-of-service forwarding-classes queue 3 priority high

set class-of-service forwarding-classes queue 4 test

set class-of-service forwarding-classes queue 4 priority high

set class-of-service interfaces ge-* scheduler-map traffic-scheduler-map

set class-of-service interfaces ge-* unit * classifiers exp traffic_classifier_p_devices

set class-of-service interfaces ge-* unit * rewrite-rules exp traffic_rewrite_rule

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class DATA_BE loss-priority low code-point 000

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VIDEO_AF loss-priority low code-point 010

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class NETWORK_CONTROL loss-priority low code-point 110

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VOICE_EF loss-priority low code-point 101

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class DATA_BE scheduler data

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VOICE_EF scheduler voice

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VIDEO_AF scheduler video

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class

NETWORK_CONTROL scheduler network

set class-of-service schedulers voice transmit-rate percent 15

set class-of-service schedulers voice buffer-size percent 0

set class-of-service schedulers voice priority high

set class-of-service schedulers video transmit-rate percent 40

set class-of-service schedulers video buffer-size percent 40

set class-of-service schedulers video priority low

set class-of-service schedulers network transmit-rate percent 10

set class-of-service schedulers network buffer-size percent 10

set class-of-service schedulers network priority high

set class-of-service schedulers data transmit-rate remainder

set class-of-service schedulers data buffer-size remainder

set class-of-service schedulers data priority low

set security forwarding-options family mpls mode packet-based

P-5

set version 12.1X47-D15.4

set system host-name P-5

set system root-authentication encrypted-password

"1HbwHNy.o$o8YQRduNh3HniuhX01WDo."

set system login user utkarsh full-name utkarsh

set system login user utkarsh uid 2000

set system login user utkarsh class super-user

set system login user utkarsh authentication encrypted-password

"1.6Rss6QQ$FKs7U/w3At08dWMLMdeWh/"

set system services ssh

set system services web-management http interface ge-0/0/0.0

set system syslog user * any emergency

set system syslog file messages any any

set system syslog file messages authorization info

set system syslog file interactive-commands interactive-commands any

set system license autoupdate url https://ae1.juniper.net/junos/key_retrieval

set interfaces ge-0/0/0 unit 0 family inet address 10.210.30.14/24

set interfaces ge-0/0/1 speed 100m

set interfaces ge-0/0/1 link-mode full-duplex

set interfaces ge-0/0/1 gigether-options no-auto-negotiation

set interfaces ge-0/0/1 unit 0 family inet address 172.32.16.50/30

set interfaces ge-0/0/1 unit 0 family mpls

set interfaces ge-0/0/2 speed 100m

set interfaces ge-0/0/2 link-mode full-duplex

set interfaces ge-0/0/2 gigether-options no-auto-negotiation

set interfaces ge-0/0/2 unit 0 family inet address 172.32.16.45/30

set interfaces ge-0/0/2 unit 0 family mpls

set interfaces ge-0/0/3 speed 100m

set interfaces ge-0/0/3 link-mode full-duplex

set interfaces ge-0/0/3 gigether-options no-auto-negotiation

set interfaces ge-0/0/3 unit 0 family inet address 172.32.16.53/30

set interfaces ge-0/0/3 unit 0 family mpls

set interfaces lo0 unit 0 family inet address 172.32.255.5/32

set interfaces lo0 unit 0 family mpls

set routing-options router-id 172.32.255.5

set routing-options autonomous-system 200

set protocols rsvp interface all link-protection

set protocols mpls icmp-tunneling

set protocols mpls interface all

set protocols ospf traffic-engineering

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface lo0.0

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service forwarding-classes queue 0 DATA_BE

set class-of-service forwarding-classes queue 0 priority low

set class-of-service forwarding-classes queue 1 VIDEO_AF

set class-of-service forwarding-classes queue 1 priority low

set class-of-service forwarding-classes queue 2 VOICE_EF

set class-of-service forwarding-classes queue 2 priority high

set class-of-service forwarding-classes queue 3 NETWORK_CONTROL

set class-of-service forwarding-classes queue 3 priority high

set class-of-service forwarding-classes queue 4 test

set class-of-service forwarding-classes queue 4 priority high

set class-of-service interfaces ge-* scheduler-map traffic-scheduler-map

set class-of-service interfaces ge-* unit * classifiers exp traffic_classifier_p_devices

set class-of-service interfaces ge-* unit * rewrite-rules exp traffic_rewrite_rule

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class DATA_BE loss-priority low code-point 000

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VIDEO_AF loss-priority low code-point 010

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class NETWORK_CONTROL loss-priority low code-point 110

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VOICE_EF loss-priority low code-point 101

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class DATA_BE scheduler data

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VOICE_EF scheduler voice

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VIDEO_AF scheduler video

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class NETWORK_CONTROL scheduler network

set class-of-service schedulers voice transmit-rate percent 15

set class-of-service schedulers voice buffer-size percent 0

set class-of-service schedulers voice priority high

set class-of-service schedulers video transmit-rate percent 40

set class-of-service schedulers video buffer-size percent 40

set class-of-service schedulers video priority low

set class-of-service schedulers network transmit-rate percent 10

set class-of-service schedulers network buffer-size percent 10

set class-of-service schedulers network priority high

set class-of-service schedulers data transmit-rate remainder

set class-of-service schedulers data buffer-size remainder

set class-of-service schedulers data priority low

set security forwarding-options family mpls mode packet-based

P-6

set version 12.1X47-D15.4

set system host-name P-6

set system root-authentication encrypted-password

"1HbwHNy.o$o8YQRduNh3HniuhX01WDo."

set system login user utkarsh full-name utkarsh

set system login user utkarsh uid 2000

set system login user utkarsh class super-user

set system login user utkarsh authentication encrypted-password

"1.6Rss6QQ$FKs7U/w3At08dWMLMdeWh/"

set system services ssh

set system services web-management http interface ge-0/0/0.0

set system syslog user * any emergency

set system syslog file messages any any

set system syslog file messages authorization info

set system syslog file interactive-commands interactive-commands any

set system license autoupdate url https://ae1.juniper.net/junos/key_retrieval

set interfaces ge-0/0/0 unit 0 family inet address 10.210.30.15/24

set interfaces ge-0/0/1 speed 100m

set interfaces ge-0/0/1 link-mode full-duplex

set interfaces ge-0/0/1 gigether-options no-auto-negotiation

set interfaces ge-0/0/1 unit 0 family inet address 172.32.16.38/30

set interfaces ge-0/0/1 unit 0 family mpls

set interfaces ge-0/0/2 speed 100m

set interfaces ge-0/0/2 link-mode full-duplex

set interfaces ge-0/0/2 gigether-options no-auto-negotiation

set interfaces ge-0/0/2 unit 0 family inet address 172.32.16.54/30

set interfaces ge-0/0/2 unit 0 family mpls

set interfaces ge-0/0/3 speed 100m

set interfaces ge-0/0/3 link-mode full-duplex

set interfaces ge-0/0/3 gigether-options no-auto-negotiation

set interfaces ge-0/0/3 unit 0 family inet address 172.32.16.65/30

set interfaces ge-0/0/3 unit 0 family mpls

set interfaces ge-0/0/4 speed 100m

set interfaces ge-0/0/4 link-mode full-duplex

set interfaces ge-0/0/4 gigether-options no-auto-negotiation

set interfaces ge-0/0/4 unit 0 family inet address 172.32.16.61/30

set interfaces ge-0/0/4 unit 0 family mpls

set interfaces lo0 unit 0 family inet address 172.32.255.6/32

set interfaces lo0 unit 0 family mpls

set routing-options router-id 172.32.255.6

set routing-options autonomous-system 200

set protocols rsvp interface all link-protection

deactivate protocols rsvp interface all link-protection

set protocols mpls icmp-tunneling

set protocols mpls no-cspf

set protocols mpls interface all

set protocols ospf traffic-engineering

deactivate protocols ospf traffic-engineering

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface lo0.0

set protocols ospf area 0.0.0.0 interface ge-0/0/4.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/4.0 bfd-liveness-detection minimum-interval 1200

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service forwarding-classes queue 0 DATA_BE

set class-of-service forwarding-classes queue 0 priority low

set class-of-service forwarding-classes queue 1 VIDEO_AF

set class-of-service forwarding-classes queue 1 priority low

set class-of-service forwarding-classes queue 2 VOICE_EF

set class-of-service forwarding-classes queue 2 priority high

set class-of-service forwarding-classes queue 3 NETWORK_CONTROL

set class-of-service forwarding-classes queue 3 priority high

set class-of-service forwarding-classes queue 4 test

set class-of-service forwarding-classes queue 4 priority high

set class-of-service interfaces ge-* scheduler-map traffic-scheduler-map

set class-of-service interfaces ge-* unit * classifiers exp traffic_classifier_p_devices

set class-of-service interfaces ge-* unit * rewrite-rules exp traffic_rewrite_rule

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class DATA_BE loss-priority low code-point 000

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VIDEO_AF loss-priority low code-point 010

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class

NETWORK_CONTROL loss-priority low code-point 110

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VOICE_EF loss-

priority low code-point 101

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class DATA_BE

scheduler data

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VOICE_EF

scheduler voice

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VIDEO_AF

scheduler video

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class

NETWORK_CONTROL scheduler network

set class-of-service schedulers voice transmit-rate percent 15

set class-of-service schedulers voice buffer-size percent 0

set class-of-service schedulers voice priority high

set class-of-service schedulers video transmit-rate percent 40

set class-of-service schedulers video buffer-size percent 40

set class-of-service schedulers video priority low

set class-of-service schedulers network transmit-rate percent 10

set class-of-service schedulers network buffer-size percent 10

set class-of-service schedulers network priority high

set class-of-service schedulers data transmit-rate remainder

set class-of-service schedulers data buffer-size remainder

set class-of-service schedulers data priority low

set security forwarding-options family mpls mode packet-based

P-7

set system host-name P-7

set system root-authentication encrypted-password

"1HbwHNy.o$o8YQRduNh3HniuhX01WDo."

set system login user utkarsh full-name utkarsh

set system login user utkarsh uid 2000

set system login user utkarsh class super-user

set system login user utkarsh authentication encrypted-password

"1.6Rss6QQ$FKs7U/w3At08dWMLMdeWh/"

set system services ssh

set system services web-management http interface ge-0/0/0.0

set system syslog user * any emergency

set system syslog file messages any any

set system syslog file messages authorization info

set system syslog file interactive-commands interactive-commands any

set system license autoupdate url https://ae1.juniper.net/junos/key_retrieval

set interfaces ge-0/0/0 unit 0 family inet address 10.210.30.16/24

set interfaces ge-0/0/1 unit 0 family inet address 172.32.16.34/30

set interfaces ge-0/0/1 unit 0 family mpls

set interfaces ge-0/0/2 unit 0 family inet address 172.32.16.42/30

set interfaces ge-0/0/2 unit 0 family mpls

set interfaces ge-0/0/3 unit 0 family inet address 172.32.16.46/30

set interfaces ge-0/0/3 unit 0 family mpls

set interfaces ge-0/0/4 unit 0 family inet address 172.32.16.85/30

set interfaces ge-0/0/4 unit 0 family mpls

set interfaces ge-0/0/5 unit 0 family inet address 172.32.16.89/30

set interfaces ge-0/0/5 unit 0 family mpls

set interfaces lo0 unit 0 family inet address 172.32.255.7/32

set interfaces lo0 unit 0 family mpls

set routing-options router-id 172.32.255.7

set routing-options autonomous-system 200

set routing-options forwarding-table export load-bal

set protocols rsvp interface all link-protection

deactivate protocols rsvp interface all link-protection

set protocols mpls icmp-tunneling

set protocols mpls no-cspf

set protocols mpls interface all

set protocols ospf traffic-engineering

deactivate protocols ospf traffic-engineering

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 node-link-protection

deactivate protocols ospf area 0.0.0.0 interface ge-0/0/2.0

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 node-link-protection

deactivate protocols ospf area 0.0.0.0 interface ge-0/0/3.0

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 node-link-protection

deactivate protocols ospf area 0.0.0.0 interface ge-0/0/1.0

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/4.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/4.0 node-link-protection

deactivate protocols ospf area 0.0.0.0 interface ge-0/0/4.0

set protocols ospf area 0.0.0.0 interface ge-0/0/4.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/5.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/5.0 node-link-protection

deactivate protocols ospf area 0.0.0.0 interface ge-0/0/5.0

set protocols ospf area 0.0.0.0 interface ge-0/0/5.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface lo0.0

set policy-options policy-statement load-bal then load-balance per-packet

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service forwarding-classes queue 0 DATA_BE

set class-of-service forwarding-classes queue 0 priority low

set class-of-service forwarding-classes queue 1 VIDEO_AF

set class-of-service forwarding-classes queue 1 priority low

set class-of-service forwarding-classes queue 2 VOICE_EF

set class-of-service forwarding-classes queue 2 priority high

set class-of-service forwarding-classes queue 3 NETWORK_CONTROL

set class-of-service forwarding-classes queue 3 priority high

set class-of-service forwarding-classes queue 4 test

set class-of-service forwarding-classes queue 4 priority high

set class-of-service interfaces ge-* scheduler-map traffic-scheduler-map

set class-of-service interfaces ge-* unit * classifiers exp traffic_classifier_p_devices

set class-of-service interfaces ge-* unit * rewrite-rules exp traffic_rewrite_rule

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class DATA_BE loss-priority low code-point 000

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VIDEO_AF loss-priority low code-point 010

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class NETWORK_CONTROL loss-priority low code-point 110

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VOICE_EF loss-priority low code-point 101

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class DATA_BE scheduler data

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VOICE_EF scheduler voice

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VIDEO_AF scheduler video

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class NETWORK_CONTROL scheduler network

set class-of-service schedulers voice transmit-rate percent 15

set class-of-service schedulers voice buffer-size percent 0

set class-of-service schedulers voice priority high

set class-of-service schedulers video transmit-rate percent 40

set class-of-service schedulers video buffer-size percent 40

set class-of-service schedulers video priority low

set class-of-service schedulers network transmit-rate percent 10

set class-of-service schedulers network buffer-size percent 10

set class-of-service schedulers network priority high

set class-of-service schedulers data transmit-rate remainder

set class-of-service schedulers data buffer-size remainder

set class-of-service schedulers data priority low

set security forwarding-options family mpls mode packet-based

P-8

set version 12.1X47-D15.4

set system host-name P-8

set system root-authentication encrypted-password

"1HbwHNy.o$o8YQRduNh3HniuhX01WDo."

set system login user utkarsh full-name utkarsh

set system login user utkarsh uid 2000

set system login user utkarsh class super-user

set system login user utkarsh authentication encrypted-password

"1.6Rss6QQ$FKs7U/w3At08dWMLMdeWh/"

set system services ssh

set system services web-management http interface ge-0/0/0.0

set system syslog user * any emergency

set system syslog file messages any any

set system syslog file messages authorization info

set system syslog file interactive-commands interactive-commands any

set system license autoupdate url https://ae1.juniper.net/junos/key_retrieval

set interfaces ge-0/0/0 unit 0 family inet address 10.210.30.17/24

set interfaces ge-0/0/1 unit 0 family inet address 172.32.16.62/30

set interfaces ge-0/0/1 unit 0 family mpls

set interfaces ge-0/0/2 unit 0 family inet address 172.32.16.73/30

set interfaces ge-0/0/2 unit 0 family mpls

set interfaces ge-0/0/3 unit 0 family inet address 172.32.16.77/30

set interfaces ge-0/0/3 unit 0 family mpls

set interfaces ge-0/0/5 unit 0 family inet address 172.32.16.61/30

set interfaces ge-0/0/5 unit 0 family mpls

set interfaces lo0 unit 0 family inet address 172.32.255.8/32

set interfaces lo0 unit 0 family mpls

set routing-options router-id 172.32.255.8

set routing-options autonomous-system 200

set protocols rsvp interface all link-protection

deactivate protocols rsvp interface all link-protection

set protocols mpls icmp-tunneling

set protocols mpls no-cspf

set protocols mpls interface all

set protocols ospf traffic-engineering

deactivate protocols ospf traffic-engineering

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 bfd-liveness-detection minimum-interval 1200

set protocols ospf area 0.0.0.0 interface lo0.0

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service forwarding-classes queue 0 DATA_BE

set class-of-service forwarding-classes queue 0 priority low

set class-of-service forwarding-classes queue 1 VIDEO_AF

set class-of-service forwarding-classes queue 1 priority low

set class-of-service forwarding-classes queue 2 VOICE_EF

set class-of-service forwarding-classes queue 2 priority high

set class-of-service forwarding-classes queue 3 NETWORK_CONTROL

set class-of-service forwarding-classes queue 3 priority high

set class-of-service forwarding-classes queue 4 test

set class-of-service forwarding-classes queue 4 priority high

set class-of-service interfaces ge-* scheduler-map traffic-scheduler-map

set class-of-service interfaces ge-* unit * classifiers exp traffic_classifier_p_devices

set class-of-service interfaces ge-* unit * rewrite-rules exp traffic_rewrite_rule

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class DATA_BE loss-priority low code-point 000

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VIDEO_AF loss-priority low code-point 010

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class NETWORK_CONTROL loss-priority low code-point 110

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VOICE_EF loss-priority low code-point 101

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class DATA_BE scheduler data

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VOICE_EF scheduler voice

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VIDEO_AF scheduler video

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class NETWORK_CONTROL scheduler network

set class-of-service schedulers voice transmit-rate percent 15

set class-of-service schedulers voice buffer-size percent 0

set class-of-service schedulers voice priority high

set class-of-service schedulers video transmit-rate percent 40

set class-of-service schedulers video buffer-size percent 40

set class-of-service schedulers video priority low

set class-of-service schedulers network transmit-rate percent 10

set class-of-service schedulers network buffer-size percent 10

set class-of-service schedulers network priority high

set class-of-service schedulers data transmit-rate remainder

set class-of-service schedulers data buffer-size remainder

set class-of-service schedulers data priority low

set security forwarding-options family mpls mode packet-based

P-9

set version 12.1X47-D15.4

set system host-name P-9

set system root-authentication encrypted-password

"1HbwHNy.o$o8YQRduNh3HniuhX01WDo."

set system login user utkarsh full-name utkarsh

set system login user utkarsh uid 2000

set system login user utkarsh class super-user

set system login user utkarsh authentication encrypted-password

"1.6Rss6QQ$FKs7U/w3At08dWMLMdeWh/"

set system services ssh

```
set system services web-management http interface ge-0/0/0.0
set system syslog user * any emergency
set system syslog file messages any any
set system syslog file messages authorization info
set system syslog file interactive-commands interactive-commands any
set system license autoupdate url https://ae1.juniper.net/junos/key_retrieval
set interfaces ge-0/0/0 unit 0 family inet address 10.210.30.18/24
set interfaces ge-0/0/1 speed 100m
set interfaces ge-0/0/1 link-mode full-duplex
set interfaces ge-0/0/1 gigether-options no-auto-negotiation
set interfaces ge-0/0/1 unit 0 family inet address 172.32.16.66/30
set interfaces ge-0/0/1 unit 0 family mpls
set interfaces ge-0/0/2 speed 100m
set interfaces ge-0/0/2 link-mode full-duplex
set interfaces ge-0/0/2 gigether-options no-auto-negotiation
set interfaces ge-0/0/2 unit 0 family inet address 172.32.16.78/30
set interfaces ge-0/0/2 unit 0 family mpls
set interfaces ge-0/0/3 speed 100m
set interfaces ge-0/0/3 link-mode full-duplex
set interfaces ge-0/0/3 gigether-options no-auto-negotiation
set interfaces ge-0/0/3 unit 0 family inet address 172.32.16.81/30
set interfaces ge-0/0/3 unit 0 family mpls
set interfaces ge-0/0/4 speed 100m
set interfaces ge-0/0/4 link-mode full-duplex
set interfaces ge-0/0/4 gigether-options no-auto-negotiation
```

set interfaces ge-0/0/4 unit 0 family inet address 172.32.16.86/30

set interfaces ge-0/0/4 unit 0 family mpls

set interfaces lo0 unit 0 family inet address 172.32.255.9/32

set interfaces lo0 unit 0 family mpls

set routing-options router-id 172.32.255.9

set routing-options autonomous-system 200

set protocols rsvp interface all link-protection

deactivate protocols rsvp interface all link-protection

set protocols mpls icmp-tunneling

set protocols mpls no-cspf

set protocols mpls interface all

set protocols ospf traffic-engineering

deactivate protocols ospf traffic-engineering

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/2.0 bfd-liveness-detection minimum-interval

1200

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/3.0 bfd-liveness-detection minimum-interval

1200

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 bfd-liveness-detection minimum-interval

1200

set protocols ospf area 0.0.0.0 interface lo0.0

set protocols ospf area 0.0.0.0 interface ge-0/0/4.0 interface-type p2p

set protocols ospf area 0.0.0.0 interface ge-0/0/4.0 bfd-liveness-detection minimum-interval 1200

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service classifiers exp traffic_classifier_p_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class DATA_BE loss-priority low code-points 000

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VIDEO_AF loss-priority low code-points 010

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class VOICE_EF loss-priority low code-points 101

set class-of-service classifiers inet-precedence traffic_classifier_pe_devices forwarding-class NETWORK_CONTROL loss-priority low code-points 110

set class-of-service forwarding-classes queue 0 DATA_BE

set class-of-service forwarding-classes queue 0 priority low

set class-of-service forwarding-classes queue 1 VIDEO_AF

set class-of-service forwarding-classes queue 1 priority low

set class-of-service forwarding-classes queue 2 VOICE_EF

set class-of-service forwarding-classes queue 2 priority high

set class-of-service forwarding-classes queue 3 NETWORK_CONTROL

set class-of-service forwarding-classes queue 3 priority high

set class-of-service forwarding-classes queue 4 test

set class-of-service forwarding-classes queue 4 priority high

set class-of-service interfaces ge-* scheduler-map traffic-scheduler-map

set class-of-service interfaces ge-* unit * classifiers exp traffic_classifier_p_devices

set class-of-service interfaces ge-* unit * rewrite-rules exp traffic_rewrite_rule

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class DATA_BE loss-priority low code-point 000

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VIDEO_AF loss-priority low code-point 010

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class NETWORK_CONTROL loss-priority low code-point 110

set class-of-service rewrite-rules exp traffic_rewrite_rule forwarding-class VOICE_EF loss-priority low code-point 101

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class DATA_BE scheduler data

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VOICE_EF scheduler voice

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class VIDEO_AF scheduler video

set class-of-service scheduler-maps traffic-scheduler-map forwarding-class NETWORK_CONTROL scheduler network

set class-of-service schedulers voice transmit-rate percent 15

set class-of-service schedulers voice buffer-size percent 0

set class-of-service schedulers voice priority high

set class-of-service schedulers video transmit-rate percent 40

set class-of-service schedulers video buffer-size percent 40

set class-of-service schedulers video priority low

set class-of-service schedulers network transmit-rate percent 10

set class-of-service schedulers network buffer-size percent 10

set class-of-service schedulers network priority high

set class-of-service schedulers data transmit-rate remainder

set class-of-service schedulers data buffer-size remainder

set class-of-service schedulers data priority low

set security forwarding-options family mpls mode packet-based